

1 RUSS, AUGUST & KABAT
2 Marc A. Fenster, SBN 181067
3 Email: mfenster@raklaw.com
4 Reza Mirzaie (CA SBN 246953)
5 Email: rmirzaie@raklaw.com
6 Brian D. Ledahl (CA SBN 186579)
7 Email: bledahl@raklaw.com
8 Paul Kroeger (CA SBN 229074)
9 Email: pkroeger@raklaw.com
10 C. Jay Chung (CA SBN 252794)
11 Email: jchung@raklaw.com
12 Philip X. Wang (CA SBN 262239)
13 Email: pwang@raklaw.com
14 12424 Wilshire Boulevard, 12th Floor
15 Los Angeles, California 90025
16 Telephone: (310) 826-7474
17 Facsimile: (310) 826-6991

18 *Attorneys for Plaintiff*

19 *REALTIME ADAPTIVE STREAMING LLC*

20
21
22
23
24
25
26
27
28
UNITED STATES DISTRICT COURT
CENTRAL DISTRICT OF CALIFORNIA
WESTERN DIVISION

REALTIME ADAPTIVE STREAMING
LLC,

Plaintiff,

vs.

GOOGLE LLC, and YOUTUBE, LLC,

Defendants.

Case No. _____

JURY TRIAL DEMANDED

COMPLAINT FOR PATENT INFRINGEMENT

This is an action for patent infringement arising under the Patent Laws of the United States of America, 35 U.S.C. § 1 *et seq.* in which Plaintiff Realtime Adaptive Streaming LLC (“Plaintiff” or “Realtime”) makes the following allegations against

1 Defendants Google LLC and YouTube, LLC. (collectively, “Defendants” or
2 “YouTube”).

3 **PARTIES**

4 1. Realtime is a Texas limited liability company. Realtime has a place of
5 business at 1828 E.S.E. Loop 323, Tyler, Texas 75701. Realtime has researched and
6 developed specific solutions for data compression. As recognition of its innovations
7 rooted in this technological field, Realtime holds multiple United States patents and
8 pending patent applications.

9 2. Defendant Google LLC is a Delaware limited liability company with its
10 principal place of business in Mountain View, California.

11 3. Defendant YouTube, LLC is a Delaware limited liability company with
12 its principal place of business in San Bruno, California. YouTube, LLC is a wholly
13 owned subsidiary of Google LLC.

14 4. Google LLC and YouTube, LLC have regular and established places of
15 business in this District, including, e.g., at 340 Main St., Venice, CA 90291. They
16 offer their products and/or services, including those accused herein of infringement, to
17 customers and potential customers located in California and in this District. Google
18 LLC and YouTube, LLC may be served with process through its registered agent for
19 service at The Corporation Service Company (CSC- Lawyers Incorporating Service),
20 2710 Gateway Oaks Drive, Suite 150N, Sacramento, California 95833.

21 **JURISDICTION AND VENUE**

22 5. This action arises under the patent laws of the United States, Title 35 of
23 the United States Code. This Court has original subject matter jurisdiction pursuant to
24 28 U.S.C. §§ 1331 and 1338(a).

25 6. This Court has personal jurisdiction over Defendants in this action
26 because they have committed acts within the Central District of California giving rise
27 to this action and has established minimum contacts with this forum such that the
28 exercise of jurisdiction over Defendants would not offend traditional notions of fair

1 play and substantial justice. Defendants have committed and continue to commit acts
2 of infringement in this District by, among other things, offering to sell and selling
3 products and/or services that infringe the asserted patents.

4 7. Venue is proper in this District, e.g., under 28 U.S.C. § 1400(b).
5 Defendants are registered to do business in California, and they have transacted
6 business in the Central District of California and have committed acts of direct and
7 indirect infringement in the Central District of California. Defendants have regular
8 and established place(s) of business in this District, as set forth above.

9
10 **THE PATENTS-IN-SUIT**

11 8. This action arises under 35 U.S.C. § 271 for Defendants' infringement of
12 Realtime's United States Patent Nos. 7,386,046 (the "'046 patent"), 8,934,535 (the
13 "'535 patent"), 9,769,477 (the "'477 patent"), 8,634,462 (the "'462 patent"), and
14 9,578,298 (the "'298 patent").

15
16 9. The '046 patent, titled "Bandwidth Sensitive Data Compression and
17 Decompression," was duly and properly issued by the United States Patent and
18 Trademark Office ("USPTO") on June 10, 2008. A copy of the '046 patent is
19 attached hereto as Exhibit A. Realtime is the owner and assignee of the '046 patent
20 and holds the right to sue for and recover all damages for infringement thereof,
21 including past infringement.
22

23
24 10. The '535 patent, titled "Systems and methods for video and audio data
25 storage and distribution," was duly and properly issued by the USPTO on January 13,
26 2015. A copy of the '535 patent is attached hereto as Exhibit B. Realtime is the
27 owner and assignee of the '535 patent and holds the right to sue for and recover all
28

1 damages for infringement thereof, including past infringement.

2 11. The '477 patent, titled "Video data compression systems," was duly and
3 properly issued by the USPTO on September 19, 2017. A copy of the '477 patent is
4 attached hereto as Exhibit C. Realtime is the owner and assignee of the '477 patent
5 and holds the right to sue for and recover all damages for infringement thereof,
6 including past infringement.
7

8 12. The '462 patent, titled "Quantization for Hybrid Video Coding," was
9 duly and properly issued by the USPTO on January 21, 2014. A copy of the '462
10 patent is attached hereto as Exhibit D. Realtime is the owner and assignee of the '462
11 patent and holds the right to sue for and recover all damages for infringement thereof,
12 including past infringement.
13

14 13. The '298 patent, titled "Method for Decoding 2D-Compatible
15 Stereoscopic Video Flows," was duly and properly issued by the USPTO on February
16 21, 2017. A copy of the '298 patent is attached hereto as Exhibit E. Realtime is the
17 owner and assignee of the '298 patent and holds the right to sue for and recover all
18 damages for infringement thereof, including past infringement.
19
20
21

22 **COUNT I**

23 **INFRINGEMENT OF U.S. PATENT NO. 7,386,046**

24 14. Plaintiff re-alleges and incorporates by reference the foregoing
25 paragraphs, as if fully set forth herein.
26

27 15. On information and belief, Defendants have made, used, offered for sale,
28

1 sold and/or imported into the United States products that infringe the '046 patent, and
2 continues to do so. By way of illustrative example, these infringing products include,
3 without limitation, Defendants' streaming products/services such as, e.g., YouTube
4 site, YouTube TV, YouTube Live Streaming Platform, YouTube App, etc., and all
5 versions and variations thereof since the issuance of the '046 patent ("Accused
6 Instrumentalities").
7

8
9 16. On information and belief, Defendants have directly infringed and continue to
10 infringe the '046 patent, for example, through their sale, offer for sale, importation,
11 use and testing of the Accused Instrumentalities, which practices the system claimed
12 by, e.g., Claim 40 of the '046 patent, namely, a system, comprising: a data
13 compression system for compressing and decompressing data input; a plurality of
14 compression routines selectively utilized by the data compression system, wherein a
15 first one of the plurality of compression routines includes a first compression
16 algorithm and a second one of the plurality of compression routines includes a second
17 compression algorithm; and a controller for tracking throughput and generating a
18 control signal to select a compression routine based on the throughput, wherein said
19 tracking throughput comprises tracking a number of pending access requests to a
20 storage device; and wherein when the controller determines that the throughput falls
21 below a predetermined throughput threshold, the controller commands the data
22 compression engine to use one of the plurality of compression routines to provide a
23 faster rate of compression so as to increase the throughput. Upon information and
24
25
26
27
28

1 belief, Defendants use the Accused Instrumentalities to practice infringing methods
2 for its own internal non-testing business purposes, while testing the Accused
3 Instrumentalities, and while providing technical support and repair services for the
4 Accused Instrumentalities to Defendants' customers. For example, the Accused
5 Instrumentalities utilize H.264 video compression standard in delivering live video
6 via HTTP Live Streaming (HLS) technology. For example, YouTube "transcodes
7 your content into lower bit rates, including HLS and iOS." *See, e.g.,*
8 <https://support.google.com/youtube/answer/6251900>. Moreover, YouTube also
9 decompresses live streams "on game consoles and mobile devices via the YouTube
10 app and m.youtube.com. *See, e.g.,* [https://support.google.com/youtube/answer/](https://support.google.com/youtube/answer/2853702?hl=en)
11 [2853702?hl=en](https://support.google.com/youtube/answer/2853702?hl=en). Furthermore, according to HLS "protocol specification does not
12 limit the encoder selection. However, the current Apple implementation should
13 interoperate with encoders that produce MPEG-2 Transport Streams containing H.264
14 video and AAC audio (HE-AAC or AAC-LC)." *See, e.g.,*
15 [https://developer.apple.com/library/content/documentation/NetworkingInternet/Conc](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html)
16 [eptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html)
17 [html](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html). As another example, HLS developer guide also states: "HTTP Live Streaming
18 supports switching between streams dynamically if the available bandwidth changes.
19 The client software uses heuristics to determine appropriate times to switch between
20 the alternates. Currently, these heuristics are based on recent trends in measured
21 network throughput." *See, e.g.,*

1 <https://developer.Apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/UsingHTTPLiveStreaming/UsingHTTPLiveStreaming.html>.

2
3
4
5 As another example, YouTube specifies ‘a plurality of compression routines
6 selectively utilized by the data compression system” as follows:

7
8 Video codec: H.264, 4.1 for up to 1080p 30 FPS
9 H.264, 4.2 for 1080p 60 FPS
10 H.264, 5.0 for 1440p 30 FPS
11 H.264, 5.1 for 1440p 60 FPS
12 H.264, 5.1 for 2160p 30 FPS
 H.264, 5.2 for 2160p 60 FPS

13 *See, e.g.,* <https://support.google.com/youtube/answer/2853702?hl=en>

14
15 Moreover, YouTube “will automatically transcode your live stream to create many
16 different output formats so all of your viewers on all of their devices and networks can
17 watch!” *See, e.g.* <https://support.google.com/youtube/answer/2853702?hl=en>.

18
19 Furthermore, HLS developer guide states: “The current implementation of the client
20 observes the effective bandwidth while playing a stream. If a higher-quality stream is
21 available and the bandwidth appears sufficient to support it, the client switches to a
22 higher quality. If a lower-quality stream is available and the current bandwidth
23 appears insufficient to support the current stream, the client switches to a lower
24 quality.” *See,*

25
26
27 *e.g.,* <https://developer.Apple.com/library/content/documentation/NetworkingInternet/>

1 Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuest
 2 ions.html.

3
 4 17. The Accused Instrumentalities include a data compression system for
 5 compressing and decompressing data input. YouTube “transcodes your content into
 6 lower bit rates, including HLS and iOS.” *See, e.g.,*
 7 [https://support.google.com/youtube](https://support.google.com/youtube/answer/6251900)
 8 [/answer/6251900](https://support.google.com/youtube/answer/6251900). For example, YouTube’s streaming products/services utilizes
 9 H.264 compression standard. As another example, YouTube’s live streaming
 10 platform utilizes H.264 compression standard.
 11
 12

13
 14
 15 Video codec: H.264, 4.1 for up to 1080p 30 FPS
 16 H.264, 4.2 for 1080p 60 FPS
 17 H.264, 5.0 for 1440p 30 FPS
 18 H.264, 5.1 for 1440p 60 FPS
 19 H.264, 5.1 for 2160p 30 FPS
 20 H.264, 5.2 for 2160p 60 FPS

21 *See, e.g.,* <https://support.google.com/youtube/answer/2853702?hl=en>.

22 18. The Accused Instrumentalities include a plurality of compression
 23 routines selectively utilized by the data compression system, wherein a first one of the
 24 plurality of compression routines includes a first compression algorithm and a second
 25 one of the plurality of compression routines includes a second compression algorithm.
 26 For example, the Accused Instrumentalities utilize H.264, which include, e.g.,
 27 Context-Adaptive Variable Length Coding (“CAVLC”) entropy encoder and Context-
 28

Adaptive Binary Arithmetic Coding (“CABAC”) entropy encoder. H.264 provides for multiple different ranges of parameters (e.g., bitrate, resolution parameters, etc.), each included in the “profiles” and “levels” defined by the H.264 standard. See http://www.axis.com/files/whitepaper/wp_h264_31669_en_0803_lo.pdf at 5:

4. H.264 profiles and levels

The joint group involved in defining H.264 focused on creating a simple and clean solution, limiting options and features to a minimum. An important aspect of the standard, as with other video standards, is providing the capabilities in profiles (sets of algorithmic features) and levels (performance classes) that optimally support popular productions and common formats.

H.264 has seven profiles, each targeting a specific class of applications. Each profile defines what feature set the encoder may use and limits the decoder implementation complexity.

Network cameras and video encoders will most likely use a profile called the baseline profile, which is intended primarily for applications with limited computing resources. The baseline profile is the most suitable given the available performance in a real-time encoder that is embedded in a network video product. The profile also enables low latency, which is an important requirement of surveillance video and also particularly important in enabling real-time, pan/tilt/zoom (PTZ) control in PTZ network cameras.

H.264 has 11 levels or degree of capability to limit performance, bandwidth and memory requirements. Each level defines the bit rate and the encoding rate in macroblock per second for resolutions ranging from QCIF to HDTV and beyond. The higher the resolution, the higher the level required.

See https://en.wikipedia.org/wiki/H.264/MPEG-4_AVC:

Levels with maximum property values

Level	Max decoding speed		Max frame size		Max video bit rate for video coding layer (VCL) kbit/s			Examples for high resolution @ highest frame rate (max stored frames) Toggle additional details
	Luma samples/s	Macroblocks/s	Luma samples	Macroblocks	Baseline, Extended and Main Profiles	High Profile	High 10 Profile	
1	380,160	1,485	25,344	99	64	80	192	176x144@15.0 (4)
1b	380,160	1,485	25,344	99	128	160	384	176x144@15.0 (4)
1.1	768,000	3,000	101,376	396	192	240	576	352x288@7.5 (2)
1.2	1,536,000	6,000	101,376	396	384	480	1,152	352x288@15.2 (6)
1.3	3,041,280	11,880	101,376	396	768	960	2,304	352x288@30.0 (6)
2	3,041,280	11,880	101,376	396	2,000	2,500	6,000	352x288@30.0 (6)
2.1	5,068,800	19,800	202,752	792	4,000	5,000	12,000	352x576@25.0 (6)
2.2	5,184,000	20,250	414,720	1,620	4,000	5,000	12,000	720x576@12.5 (5)
3	10,368,000	40,500	414,720	1,620	10,000	12,500	30,000	720x576@25.0 (5)
3.1	27,648,000	108,000	921,600	3,600	14,000	17,500	42,000	1,280x720@30.0 (5)
3.2	55,296,000	216,000	1,310,720	5,120	20,000	25,000	60,000	1,280x1,024@42.2 (4)
4	62,914,560	245,760	2,097,152	8,192	20,000	25,000	60,000	2,048x1,024@30.0 (4)
4.1	62,914,560	245,760	2,097,152	8,192	50,000	62,500	150,000	2,048x1,024@30.0 (4)
4.2	133,693,440	522,240	2,228,224	8,704	50,000	62,500	150,000	2,048x1,080@60.0 (4)
5	150,994,944	589,824	5,652,480	22,080	135,000	168,750	405,000	3,672x1,536@26.7 (5)
5.1	251,658,240	983,040	9,437,184	36,864	240,000	300,000	720,000	4,096x2,304@26.7 (5)
5.2	530,841,600	2,073,600	9,437,184	36,864	240,000	300,000	720,000	4,096x2,304@56.3 (5)

19. A video data block is organized by the group of pictures (GOP) structure,

1 which is a “collection of successive pictures within a coded video stream.” *See*
2 https://en.wikipedia.org/wiki/Group_of_pictures. A GOP structure can contain intra
3 coded pictures (I picture or I frame), predictive coded pictures (P picture or P frame),
4 bipredictive coded pictures (B picture or B frame) and direct coded pictures (D picture
5 or D frames, or DC direct coded pictures which are used only in MPEG-1 video). *See*
6 https://en.wikipedia.org/wiki/Video_compression_picture_types (for descriptions of I
7 frames, P frames and B frames); <https://en.wikipedia.org/wiki/MPEG-1#D-frames> (for
8 descriptions of D frames). Thus, at least a portion of a video data block would also
9 make up a GOP structure and could also contain I frames, P frames, B frames and/or
10 D frames. The GOP structure also reflects the size of a video data block, and the GOP
11 structure can be controlled and used to fine-tune other parameters (e.g. bitrate, max
12 video bitrate and resolution parameters) or even be considered as a parameter by itself.

13
14 20. Based on the bitrate and/or resolution parameter identified (e.g. bitrate,
15 max video bitrate, resolution, GOP structure or frame type within a GOP structure), a
16 H.264-compliant system such as the Accused Instrumentalities would determine
17 which profile (e.g., “baseline,” “extended,” “main”, or “high”) corresponds with that
18 parameter, then select between at least two asymmetric compressors. If baseline or
19 extended is the corresponding profile, then the system will select a Context-Adaptive
20 Variable Length Coding (“CAVLC”) entropy encoder. If main or high is the
21 corresponding profile, then the system will select a Context-Adaptive Binary
22 Arithmetic Coding (“CABAC”) entropy encoder. *See*
23
24
25
26
27
28

<https://sonnati.wordpress.com/2007/10/29/how-h-264-works-part-ii/>

	Baseline	Extended	Main	High	High 10
I and P Slices	Yes	Yes	Yes	Yes	Yes
B Slices	No	Yes	Yes	Yes	Yes
SI and SP Slices	No	Yes	No	No	No
Multiple Reference Frames	Yes	Yes	Yes	Yes	Yes
In-Loop Deblocking Filter	Yes	Yes	Yes	Yes	Yes
CAVLC Entropy Coding	Yes	Yes	Yes	Yes	Yes
CABAC Entropy Coding	No	No	Yes	Yes	Yes
Flexible Macroblock Ordering (FMO)	Yes	Yes	No	No	No
Arbitrary Slice Ordering (ASO)	Yes	Yes	No	No	No
Redundant Slices (RS)	Yes	Yes	No	No	No
Data Partitioning	No	Yes	No	No	No
Interlaced Coding (PicAFF, MBAFF)	No	Yes	Yes	Yes	Yes
4:2:0 Chroma Format	Yes	Yes	Yes	Yes	Yes
Monochrome Video Format (4:0:0)	No	No	No	Yes	Yes
4:2:2 Chroma Format	No	No	No	No	No
4:4:4 Chroma Format	No	No	No	No	No
8 Bit Sample Depth	Yes	Yes	Yes	Yes	Yes
9 and 10 Bit Sample Depth	No	No	No	No	Yes
11 to 14 Bit Sample Depth	No	No	No	No	No
8x8 vs. 4x4 Transform Adaptivity	No	No	No	Yes	Yes
Quantization Scaling Matrices	No	No	No	Yes	Yes
Separate Cb and Cr QP control	No	No	No	Yes	Yes
Separate Color Plane Coding	No	No	No	No	No
Predictive Lossless Coding	No	No	No	No	No

See http://web.cs.ucla.edu/classes/fall03/cs218/paper/H.264_MPEG4_Tutorial.pdf at 7:

The following table summarizes the two major types of entropy coding: Variable Length Coding (VLC) and Context Adaptive Binary Arithmetic Coding (CABAC). CABAC offers superior coding efficiency over VLC by adapting to the changing probability distribution of symbols, by exploiting correlation between symbols, and by adaptively exploiting bit correlations using arithmetic coding. H.264 also supports Context Adaptive Variable Length Coding (CAVLC) which offers superior entropy coding over VLC without the full cost of CABAC.

H.264 Entropy Coding – Comparison of Approaches

Characteristics	Variable Length Coding (VLC)	Context Adaptive Binary Arithmetic Coding(CABAC)
• Where it is used	MPEG-2, MPEG-4 ASP	H.264/MPEG-4 AVC (high efficiency option)
• Probability distribution	Static - Probabilities never change	Adaptive - Adjusts probabilities based on actual data
• Leverages correlation between symbols	No - Conditional probabilities ignored	Yes - Exploits symbol correlations by using "contexts"
• Non-integer code words	No - Low coding efficiency for high probability symbols	Yes - Exploits "arithmetic coding" which generates non-integer code words for higher efficiency

Moreover, the H.264 Standard requires a bit-flag descriptor, which is set to determine the correct decoder for the corresponding encoder. As shown below, if the flag = 0, then CAVLC must have been selected as the encoder; if the flag = 1, then CABAC must have been selected as the encoder. *See*

https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-H.264-201304-S!!PDF-E&type=items (Rec. ITU-T H.264 (04/2013)) at 80:

entropy_coding_mode_flag selects the entropy decoding method to be applied for the syntax elements for which two descriptors appear in the syntax tables as follows:

- If **entropy_coding_mode_flag** is equal to 0, the method specified by the left descriptor in the syntax table is applied (Exp-Golomb coded, see clause 9.1 or CAVLC, see clause 9.2).
- Otherwise (**entropy_coding_mode_flag** is equal to 1), the method specified by the right descriptor in the syntax table is applied (CABAC, see clause 9.3).

21. After its selection, the asymmetric compressor (CAVLC or CABAC) will compress the video data to provide various compressed data blocks, which can be organized in a GOP structure (see above). *See* <https://sonnati.wordpress.com/2007/10/29/how-h-264-works-part-ii/>:

Entropy Coding

For entropy coding, H.264 may use an enhanced VLC, a more complex context-adaptive variable-length coding (CAVLC) or an ever more complex Context-adaptive binary-arithmetic coding (CABAC) which are complex techniques to losslessly compress syntax elements in the video stream knowing the probabilities of syntax elements in a given context. The use of CABAC can improve the compression of around 5-7%. CABAC may requires a 30-40% of total processing power to be accomplished.

22. *See*

<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.602.1581&rep=rep1&type=pdf> at 13:

Typical compression ratios to maintain excellent quality are:

- 10:1 for general images using JPEG
- 30:1 for general video using H.263 and MPEG-2
- 60:1 for general video using H.264 and WMV9

See http://www.ijera.com/papers/Vol3_issue4/BM34399403.pdf at 2:

1 Most visual communication systems today
2 use Baseline Profile. Baseline is the simplest H.264
3 profile and defines, for example, zigzag scanning of
4 the picture and using 4:2:0 (YUV video formats)
5 chrominance sampling. In Baseline Profile, the
6 picture is split in blocks consisting of 4x4 pixels,
7 and each block is processed separately. Another
8 important element of the Baseline Profile is the use
9 of Universal Variable Length Coding (UVLC) and
10 Context Adaptive Variable Length Coding
11 (CAVLC) entropy coding techniques.

12 The Extended and Main Profiles includes
13 the functionality of the Baseline Profile and add
14 improvements to the predictions algorithms. Since
15 transmitting every single frame (think 30 frames per
16 second for good quality video) is not feasible if you
17 are trying to reduce the bit rate 1000-2000 times,
18 temporal and motion prediction are heavily used in
19 H.264, and allow transmitting only the difference
20 between one frame and the previous frames. The
21 result is spectacular efficiency gain, especially for
22 scenes with little change and motion.

23 The High Profile is the most powerful
24 profile in H.264, and it allows most efficient coding
25 of video. For example, large coding gain achieved
26 through the use of Context Adaptive Binary
27 Arithmetic Coding (CABAC) encoding which is
28 more efficient than the UVLC/CAVLC used in
Baseline Profile.

The High Profile also uses adaptive
transform that decides on the fly if 4x4 or 8x8-pixel
blocks should be used. For example, 4x4 blocks are
used for the parts of the picture that are dense with
detail, while parts that have little detail are
transformed using 8x8 blocks.

23. The Accused Instrumentalities includes a controller for tracking
throughput and generating a control signal to select a compression routine based on
the throughput, wherein said tracking throughput comprises tracking a number of
pending access requests to a storage device, and a controller where, when the
controller determines that the throughput falls below a predetermined throughput
threshold, the controller commands the data compression engine to use one of the
plurality of compression routines to provide a faster rate of compression so as to

1 increase the throughput. For example, YouTube “will automatically transcode your
2 live stream to create many different output formats so all of your viewers on all of
3 their devices and networks can watch!”

4
5 *See, e.g.* <https://support.google.com/youtube/answer/2853702?hl=en>.

6 In this regard, YouTube “transcodes your content into lower bit rates, including HLS
7 and iOS.” *See, e.g.*, <https://support.google.com/youtube/answer/6251900>. As such,
8
9 HLS (HTTP Live Streaming) “supports switching between streams dynamically if the
10 available bandwidth changes. The client software uses heuristics to determine
11 appropriate times to switch between the alternates. Currently, these heuristics are
12 based on recent trends in measured network throughput.” *See, e.g.*,
13
14 [https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/UsingHTTPLiveStreaming/UsingHTTPLiveStreaming.](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/UsingHTTPLiveStreaming/UsingHTTPLiveStreaming.html)
15
16 [html](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/UsingHTTPLiveStreaming/UsingHTTPLiveStreaming.html). “The current implementation of the client observes the effective bandwidth
17 while playing a stream. If a higher-quality stream is available and the bandwidth
18 appears sufficient to support it, the client switches to a higher quality. If a lower-
19 quality stream is available and the current bandwidth appears insufficient to support
20 the current stream, the client switches to a lower quality.” *See, e.g.*,
21
22 [https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html)
23
24 [html](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html). The controller in the Accused Instrumentalities decides which compression (e.g.,
25
26 CABAC, CAVLC, etc.) to use at a point in time based on parameters, for example,
27
28

1 e.g., current or anticipated throughput. For example, when a low bandwidth is present,
2 the Accused Instrumentalities select lower quality stream using a particular
3 compression technique. As another example, when a high bandwidth is present, the
4 Accused Instrumentalities select higher quality stream using another particular
5 compression technique. For example, the Accused Instrumentalities' use of HTTP
6 Live Streaming is directed to this selection. As another example, the Accused
7 Instrumentalities' use of different "Profiles" of H.264 is directed to selecting lower
8 quality stream using a particular compression technique (e.g., CABAC or CAVLC,
9 etc.) for lower anticipated bandwidth situations, and selecting higher quality stream
10 using a higher compression technique (e.g., CABAC or CAVLC, etc.) for higher
11 anticipated bandwidth situations.
12
13
14

15 24. On information and belief, Defendants also directly infringe and continue
16 to infringe other claims of the '046 patent.
17

18 25. On information and belief, all of the Accused Instrumentalities perform
19 the claimed methods in substantially the same way, e.g., in the manner specified in the
20 H.264 standard.
21

22 26. On information and belief, use of the Accused Instrumentalities in their
23 ordinary and customary fashion results in infringement of the methods and systems
24 claimed by the '046 patent.
25

26 27. On information and belief, Defendants have had knowledge of the '046
27 patent since at least the filing of this Complaint or shortly thereafter, and on
28

1 information and belief, Defendants knew of the '046 patent and knew of its
2 infringement, including by way of this lawsuit. By the time of trial, Defendants will
3 have known and intended (since receiving such notice) that its continued actions
4 would actively induce and contribute to the infringement of the claims of the '046
5 patent.
6

7 28. Upon information and belief, Defendants' affirmative acts of making,
8 using, and selling the Accused Instrumentalities, and providing implementation
9 services and technical support to users of the Accused Instrumentalities, including,
10 e.g., through training, demonstrations, brochures, installation and user guides, have
11 induced and continue to induce users of the Accused Instrumentalities to use them in
12 their normal and customary way to infringe the '046 patent. For example, Defendants
13 adopted H.264 as its video codec in its YouTube TV, YouTube Live Streaming
14 Platform, and YouTube App. As another example, YouTube on its help webpages
15 states that "[A]n **encoder** compresses audio and video into a format that can be
16 delivered to the YouTube platform. YouTube transcodes your content into lower bit
17 rates, including HLS and iOS. This means your content will be available to as wide
18 an audience as possible on desktop, mobile and tablet."
19
20
21
22

23 *See, e.g.* <https://support.google.com/youtube/answer/6251900>. For similar reasons,
24 Defendants also induce their customers to use the Accused Instrumentalities to
25 infringe other claims of the '046 patent. Defendants specifically intended and was
26 aware that these normal and customary activities would infringe the '046 patent.
27
28

1 Defendants performed the acts that constitute induced infringement, and would induce
2 actual infringement, with the knowledge of the '046 patent and with the knowledge, or
3 willful blindness to the probability, that the induced acts would constitute
4 infringement. For example, since filing of this action, Defendants know that the
5 ordinary way of using HTTP Live Streaming—which is directed to choosing different
6 compression techniques based on current or anticipated throughput—in the Accused
7 Instrumentalities infringes the patent but nevertheless continues to promote HTTP
8 Live Streaming to customers. The only reasonable inference is that Defendants
9 specifically intend the users to infringe the patent. On information and belief,
10 Defendants engaged in such inducement to promote the sales of the Accused
11 Instrumentalities. Accordingly, Defendants have induced and continue to induce users
12 of the Accused Instrumentalities to use the Accused Instrumentalities in their ordinary
13 and customary way to infringe the '046 patent, knowing that such use constitutes
14 infringement of the '046 patent. Accordingly, Defendants have been (as of filing of
15 the original complaint), and currently are, inducing infringement of the '046 patent, in
16 violation of 35 U.S.C. § 271(b).

22 29. Defendants have also infringed, and continue to infringe, claims of the
23 '046 patent by offering to commercially distribute, commercially distributing, making,
24 and/or importing the Accused Instrumentalities, which are used in practicing the
25 process, or using the systems, of the '046 patent, and constitute a material part of the
26 invention. Defendants know the components in the Accused Instrumentalities to be
27
28

1 especially made or especially adapted for use in infringement of the '046 patent, not a
2 staple article, and not a commodity of commerce suitable for substantial noninfringing
3 use. For example, the ordinary way of using HTTP Live Streaming—which is
4 directed to choosing different compression techniques based on current or anticipated
5 throughput—infringes the patent, and as such, is especially adapted for use in
6 infringement. Moreover, there is no substantial noninfringing use, as HTTP Live
7 Streaming is directed to choosing different compression techniques based on current
8 or anticipated throughput. Accordingly, Defendants have been (as of filing of the
9 original complaint), and currently is, contributorily infringing the '046 patent, in
10 violation of 35 U.S.C. § 271(c).

14 30. By making, using, offering for sale, selling and/or importing into the
15 United States the Accused Instrumentalities, and touting the benefits of using the
16 Accused Instrumentalities' compression features, Defendants have injured Realtime
17 and is liable to Realtime for infringement of the '046 patent pursuant to 35 U.S.C. §
18 271.
19

21 31. As a result of Defendants' infringement of the '046 patent, Plaintiff
22 Realtime is entitled to monetary damages in an amount adequate to compensate for
23 Defendants' infringement, but in no event less than a reasonable royalty for the use
24 made of the invention by Defendants, together with interest and costs as fixed by the
25 Court.
26

27 **COUNT II**

28 **INFRINGEMENT OF U.S. PATENT NO. 8,934,535**

1 32. Plaintiff re-alleges and incorporates by reference the foregoing
2 paragraphs, as if fully set forth herein.

3 33. On information and belief, Defendants have made, used, offered for sale,
4 sold and/or imported into the United States products that infringe the '535 patent, and
5 continues to do so. By way of illustrative example, these infringing products include,
6 without limitation, YouTube's streaming products/services such as, e.g., YouTube
7 site, YouTube TV, YouTube Live Streaming Platform, YouTube App, etc., and all
8 versions and variations thereof since the issuance of the '535 patent ("Accused
9 Instrumentalities").
10
11

12 34. On information and belief, Defendants have directly infringed and
13 continues to infringe the '535 patent, for example, through its own use and testing of
14 the Accused Instrumentalities, which when used, practices the method claimed by
15 Claim 15 of the '535 patent, namely, a method, comprising: determining a parameter
16 of at least a portion of a data block; selecting one or more asymmetric compressors
17 from among a plurality of compressors based upon the determined parameter or
18 attribute; compressing the at least the portion of the data block with the selected one
19 or more asymmetric compressors to provide one or more compressed data blocks; and
20 storing at least a portion of the one or more compressed data blocks. Upon
21 information and belief, Defendants use the Accused Instrumentalities to practice
22 infringing methods for its own internal non-testing business purposes, while testing
23 the Accused Instrumentalities, and while providing technical support and repair
24
25
26
27
28

1 services for the Accused Instrumentalities to Defendants' customers.

2 35. For example, the Accused Instrumentalities utilize H.264 video
3 compression standard in delivering live video via HTTP Live Streaming (HLS)
4 technology. For example, YouTube "transcodes your content into lower bit rates,
5 including HLS and iOS." *See, e.g.,*

6 <https://support.google.com/youtube/answer/6251900>. Furthermore, according to
7 HLS "protocol specification does not limit the encoder selection. However, the current
8 Apple implementation should interoperate with encoders that produce MPEG-2
9 Transport Streams containing H.264 video and AAC audio (HE-AAC or AAC-LC)."
10 *See, e.g.,*

11 [https://developer.apple.com/library/content/documentation/NetworkingInternet/Conc
12 eptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.
13 html](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html). As another example, HLS developer guide also states: "HTTP Live Streaming
14 supports switching between streams dynamically if the available bandwidth changes.
15 The client software uses heuristics to determine appropriate times to switch between
16 the alternates. Currently, these heuristics are based on recent trends in measured
17 network throughput." *See, e.g.,*

18 [https://developer.apple.com/library/content/documentation/NetworkingInternet/Conc
19 eptual/StreamingMediaGuide/UsingHTTPLiveStreaming/UsingHTTPLiveStreaming.
20 html](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/UsingHTTPLiveStreaming/UsingHTTPLiveStreaming.html). Moreover, HLS developer guide states: "The current implementation of the
21 client observes the effective bandwidth while playing a stream. If a higher-quality
22

1 stream is available and the bandwidth appears sufficient to support it, the client
2 switches to a higher quality. If a lower-quality stream is available and the current
3 bandwidth appears insufficient to support the current stream, the client switches to a
4 lower quality.” *See,*
5 *e.g.,* [https://developer.Apple.com/library/content/documentation/NetworkingInternet/
6 Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuest
7 ions.html](https://developer.Apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html).
8
9

10 36. The Accused Instrumentalities determine a parameter of at least a portion
11 of a video data block. For example, YouTube “automatically detects the stream
12 resolution and frame rate. When you are live, we’ll transcode to lower resolutions so
13 all of your fans can enjoy your stream no matter the quality of their Internet
14 connection.” *See,* *e.g.,*

15 https://support.google.com/youtube/answer/2853700?hl=en&ref_topic=6136989
16

17 As shown below, examples of such parameters include bitrate (or max video bitrate)
18 and resolution parameters. Different parameters correspond with different end
19 applications. H.264 provides for multiple different ranges of such parameters, each
20 included in the “profiles” and “levels” defined by the H.264 standard. *See*
21 http://www.axis.com/files/whitepaper/wp_h264_31669_en_0803_lo.pdf at 5:
22
23
24
25
26
27
28

4. H.264 profiles and levels

The joint group involved in defining H.264 focused on creating a simple and clean solution, limiting options and features to a minimum. An important aspect of the standard, as with other video standards, is providing the capabilities in profiles (sets of algorithmic features) and levels (performance classes) that optimally support popular productions and common formats.

H.264 has seven profiles, each targeting a specific class of applications. Each profile defines what feature set the encoder may use and limits the decoder implementation complexity.

Network cameras and video encoders will most likely use a profile called the baseline profile, which is intended primarily for applications with limited computing resources. The baseline profile is the most suitable given the available performance in a real-time encoder that is embedded in a network video product. The profile also enables low latency, which is an important requirement of surveillance video and also particularly important in enabling real-time, pan/tilt/zoom (PTZ) control in PTZ network cameras.

H.264 has 11 levels or degree of capability to limit performance, bandwidth and memory requirements. Each level defines the bit rate and the encoding rate in macroblock per second for resolutions ranging from QCIF to HDTV and beyond. The higher the resolution, the higher the level required.

See https://en.wikipedia.org/wiki/H.264/MPEG-4_AVC:

Levels with maximum property values

Level	Max decoding speed		Max frame size		Max video bit rate for video coding layer (VCL) kbit/s			Examples for high resolution @ highest frame rate (max stored frames) Toggle additional details
	Luma samples/s	Macroblocks/s	Luma samples	Macroblocks	Baseline, Extended and Main Profiles	High Profile	High 10 Profile	
1	380,160	1,485	25,344	99	64	80	192	176x144@15.0 (4)
1b	380,160	1,485	25,344	99	128	160	384	176x144@15.0 (4)
1.1	768,000	3,000	101,376	396	192	240	576	352x288@7.5 (2)
1.2	1,536,000	6,000	101,376	396	384	480	1,152	352x288@15.2 (6)
1.3	3,041,280	11,880	101,376	396	768	960	2,304	352x288@30.0 (6)
2	3,041,280	11,880	101,376	396	2,000	2,500	6,000	352x288@30.0 (6)
2.1	5,068,800	19,800	202,752	792	4,000	5,000	12,000	352x576@25.0 (6)
2.2	5,184,000	20,250	414,720	1,620	4,000	5,000	12,000	720x576@12.5 (5)
3	10,368,000	40,500	414,720	1,620	10,000	12,500	30,000	720x576@25.0 (5)
3.1	27,648,000	108,000	921,600	3,600	14,000	17,500	42,000	1,280x720@30.0 (5)
3.2	55,296,000	216,000	1,310,720	5,120	20,000	25,000	60,000	1,280x1,024@42.2 (4)
4	62,914,560	245,760	2,097,152	8,192	20,000	25,000	60,000	2,048x1,024@30.0 (4)
4.1	62,914,560	245,760	2,097,152	8,192	50,000	62,500	150,000	2,048x1,024@30.0 (4)
4.2	133,693,440	522,240	2,228,224	8,704	50,000	62,500	150,000	2,048x1,080@60.0 (4)
5	150,994,944	589,824	5,652,480	22,080	135,000	168,750	405,000	3,672x1,536@26.7 (5)
5.1	251,658,240	983,040	9,437,184	36,864	240,000	300,000	720,000	4,096x2,304@26.7 (5)
5.2	530,841,600	2,073,600	9,437,184	36,864	240,000	300,000	720,000	4,096x2,304@56.3 (5)

37. A video data block is organized by the group of pictures (GOP) structure, which is a “collection of successive pictures within a coded video stream.” See https://en.wikipedia.org/wiki/Group_of_pictures. A GOP structure can contain intra coded pictures (I picture or I frame), predictive coded pictures (P picture or P frame), bipredictive coded pictures (B picture or B frame) and direct coded pictures (D picture

1 or D frames, or DC direct coded pictures which are used only in MPEG-1 video). *See*
2 https://en.wikipedia.org/wiki/Video_compression_picture_types (for descriptions of I
3 frames, P frames and B frames); <https://en.wikipedia.org/wiki/MPEG-1#D-frames> (for
4 descriptions of D frames). Thus, at least a portion of a video data block would also
5 make up a GOP structure and could also contain I frames, P frames, B frames and/or
6 D frames. The GOP structure also reflects the size of a video data block, and the GOP
7 structure can be controlled and used to fine-tune other parameters (e.g. bitrate, max
8 video bitrate and resolution parameters) or even be considered as a parameter by itself.
9

11 38. Based on the bitrate and/or resolution parameter identified (e.g. bitrate,
12 max video bitrate, resolution, GOP structure or frame type within a GOP structure),
13 any H.264-compliant system such as the Accused Instrumentalities would determine
14 which profile (e.g., “baseline,” “extended,” “main”, or “high”) corresponds with that
15 parameter, then select between at least two asymmetric compressors. If baseline or
16 extended is the corresponding profile, then the system will select a Context-Adaptive
17 Variable Length Coding (“CAVLC”) entropy encoder. If main or high is the
18 corresponding profile, then the system will select a Context-Adaptive Binary
19 Arithmetic Coding (“CABAC”) entropy encoder. Both encoders are asymmetric
20 compressors because it takes a longer period of time for them to compress data than to
21 decompress data. *See* [https://sonnati.wordpress.com/2007/10/29/how-h-264-works-](https://sonnati.wordpress.com/2007/10/29/how-h-264-works-part-ii/)
22 [part-ii/](https://sonnati.wordpress.com/2007/10/29/how-h-264-works-part-ii/)
23
24
25
26
27
28

	Baseline	Extended	Main	High	High 10
I and P Slices	Yes	Yes	Yes	Yes	Yes
B Slices	No	Yes	Yes	Yes	Yes
SI and SP Slices	No	Yes	No	No	No
Multiple Reference Frames	Yes	Yes	Yes	Yes	Yes
In-Loop Deblocking Filter	Yes	Yes	Yes	Yes	Yes
CAVLC Entropy Coding	Yes	Yes	Yes	Yes	Yes
CABAC Entropy Coding	No	No	Yes	Yes	Yes
Flexible Macroblock Ordering (FMO)	Yes	Yes	No	No	No
Arbitrary Slice Ordering (ASO)	Yes	Yes	No	No	No
Redundant Slices (RS)	Yes	Yes	No	No	No
Data Partitioning	No	Yes	No	No	No
Interlaced Coding (PicAFF, MBAFF)	No	Yes	Yes	Yes	Yes
4:2:0 Chroma Format	Yes	Yes	Yes	Yes	Yes
Monochrome Video Format (4:0:0)	No	No	No	Yes	Yes
4:2:2 Chroma Format	No	No	No	No	No
4:4:4 Chroma Format	No	No	No	No	No
8 Bit Sample Depth	Yes	Yes	Yes	Yes	Yes
9 and 10 Bit Sample Depth	No	No	No	No	Yes
11 to 14 Bit Sample Depth	No	No	No	No	No
8x8 vs. 4x4 Transform Adaptivity	No	No	No	Yes	Yes
Quantization Scaling Matrices	No	No	No	Yes	Yes
Separate Cb and Cr QP control	No	No	No	Yes	Yes
Separate Color Plane Coding	No	No	No	No	No
Predictive Lossless Coding	No	No	No	No	No

See http://web.cs.ucla.edu/classes/fall03/cs218/paper/H.264_MPEG4_Tutorial.pdf at 7:

The following table summarizes the two major types of entropy coding: Variable Length Coding (VLC) and Context Adaptive Binary Arithmetic Coding (CABAC). CABAC offers superior coding efficiency over VLC by adapting to the changing probability distribution of symbols, by exploiting correlation between symbols, and by adaptively exploiting bit correlations using arithmetic coding. H.264 also supports Context Adaptive Variable Length Coding (CAVLC) which offers superior entropy coding over VLC without the full cost of CABAC.

H.264 Entropy Coding – Comparison of Approaches

Characteristics	Variable Length Coding (VLC)	Context Adaptive Binary Arithmetic Coding(CABAC)
• Where it is used	MPEG-2, MPEG-4 ASP	H.264/MPEG-4 AVC (high efficiency option)
• Probability distribution	Static - Probabilities never change	Adaptive - Adjusts probabilities based on actual data
• Leverages correlation between symbols	No - Conditional probabilities ignored	Yes - Exploits symbol correlations by using "contexts"
• Non-integer code words	No - Low coding efficiency for high probability symbols	Yes - Exploits "arithmetic coding" which generates non-integer code words for higher efficiency

Moreover, the H.264 Standard requires a bit-flag descriptor, which is set to determine the correct decoder for the corresponding encoder. As shown below, if the flag = 0, then CAVLC must have been selected as the encoder; if the flag = 1, then CABAC must have been selected as the encoder. *See* https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-H.264-201304-S!!PDF-E&type=items (Rec. ITU-T H.264 (04/2013)) at 80:

entropy_coding_mode_flag selects the entropy decoding method to be applied for the syntax elements for which two descriptors appear in the syntax tables as follows:

- If **entropy_coding_mode_flag** is equal to 0, the method specified by the left descriptor in the syntax table is applied (Exp-Golomb coded, see clause 9.1 or CAVLC, see clause 9.2).
- Otherwise (**entropy_coding_mode_flag** is equal to 1), the method specified by the right descriptor in the syntax table is applied (CABAC, see clause 9.3).

The controller in the Accused Instrumentalities decides which compression (e.g., CABAC, CAVLC, etc.) to use at a point in time based on parameters, for example,

e.g., current or anticipated throughput. For example, when a low bandwidth is present, the Accused Instrumentalities select lower quality stream using a particular compression technique. As another example, when a high bandwidth is present, the Accused Instrumentalities select higher quality stream using another particular compression technique. For example, the Accused Instrumentalities' use of HTTP Live Streaming is directed to this selection. As another example, the Accused Instrumentalities' use of different "Profiles" of H.264 is directed to selecting lower quality stream using a particular compression technique (e.g., CABAC or CAVLC, etc.) for lower anticipated bandwidth situations, and selecting higher quality stream using a higher compression technique (e.g., CABAC or CAVLC, etc.) for higher anticipated bandwidth situations.

39. Moreover, compression techniques utilized in H.264 standard are asymmetric.

Network and Complex Systems
ISSN 2224-610X (Paper) ISSN 2225-0603 (Online)
Vol.4, No.2, 2014

www.iiste.org
IISTE

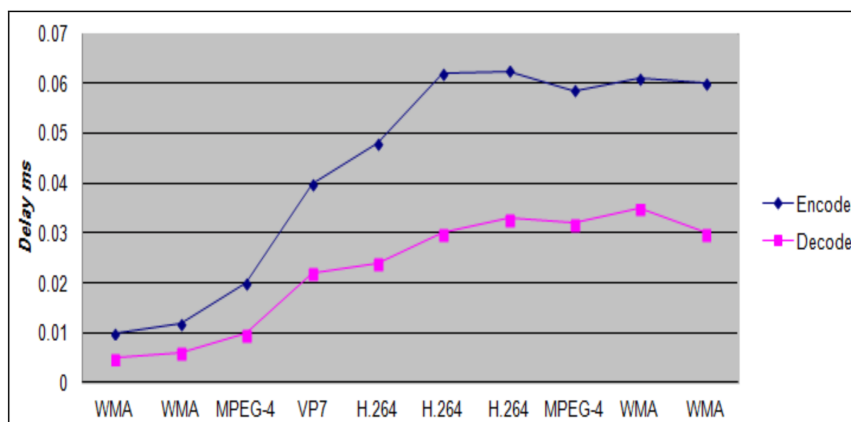


Figure 4: Encoder and Decoder Delay for Variety of Video Codec's

It is clear from figure 4, decoder delay in the most of video codec's can be assumed half of the encode delay.

See e.g., <http://www.iiste.org/Journals/index.php/NCS/article/viewFile/11072/11373>

40. YouTube "will automatically transcode your live stream to create many

different output formats so all of your viewers on all of their devices and networks can watch!” See, e.g. <https://support.google.com/youtube/answer/2853702?hl=en>.

In particular, the Accused Instrumentalities compress the at least the portion of the data block with the selected one or more asymmetric compressors to provide one or more compressed data blocks, which can be organized in a GOP structure (see above).

After its selection, the asymmetric compressor (CAVLC or CABAC) will compress the video data to provide various compressed data blocks, which can also be organized in a GOP structure. See <https://sonnati.wordpress.com/2007/10/29/how-h-264-works-part-ii/>:

Entropy Coding

For entropy coding, H.264 may use an enhanced VLC, a more complex context-adaptive variable-length coding (CAVLC) or an ever more complex Context-adaptive binary-arithmetic coding (CABAC) which are complex techniques to losslessly compress syntax elements in the video stream knowing the probabilities of syntax elements in a given context. The use of CABAC can improve the compression of around 5-7%. CABAC may requires a 30-40% of total processing power to be accomplished.

41. See

<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.602.1581&rep=rep1&type=pdf> at 13:

Typical compression ratios to maintain excellent quality are:

- 10:1 for general images using JPEG
- 30:1 for general video using H.263 and MPEG-2
- 60:1 for general video using H.264 and WMV9

See http://www.ijera.com/papers/Vol3_issue4/BM34399403.pdf at 2:

1 Most visual communication systems today
2 use Baseline Profile. Baseline is the simplest H.264
3 profile and defines, for example, zigzag scanning of
4 the picture and using 4:2:0 (YUV video formats)
5 chrominance sampling. In Baseline Profile, the
6 picture is split in blocks consisting of 4x4 pixels,
7 and each block is processed separately. Another
8 important element of the Baseline Profile is the use
9 of Universal Variable Length Coding (UVLC) and
10 Context Adaptive Variable Length Coding
11 (CAVLC) entropy coding techniques.

12 The Extended and Main Profiles includes
13 the functionality of the Baseline Profile and add
14 improvements to the predictions algorithms. Since
15 transmitting every single frame (think 30 frames per
16 second for good quality video) is not feasible if you
17 are trying to reduce the bit rate 1000-2000 times,
18 temporal and motion prediction are heavily used in
19 H.264, and allow transmitting only the difference
20 between one frame and the previous frames. The
21 result is spectacular efficiency gain, especially for
22 scenes with little change and motion.

23 The High Profile is the most powerful
24 profile in H.264, and it allows most efficient coding
25 of video. For example, large coding gain achieved
26 through the use of Context Adaptive Binary
27 Arithmetic Coding (CABAC) encoding which is
28 more efficient than the UVLC/CAVLC used in
Baseline Profile.

The High Profile also uses adaptive
transform that decides on the fly if 4x4 or 8x8-pixel
blocks should be used. For example, 4x4 blocks are
used for the parts of the picture that are dense with
detail, while parts that have little detail are
transformed using 8x8 blocks.

42. YouTube will “automatically archive the event up to 12 hours and make
it available in the Video Manager.” *See, e.g.*

https://support.google.com/youtube/answer/2853700?hl=en&ref_topic=6136989

Thus, on information and belief, the Accused Instrumentalities store at least a portion
of the one or more compressed data blocks in buffers, hard disk, or other forms of
memory/storage.

1 43. On information and belief, Defendants also directly infringe and continue
2 to infringe other claims of the '535 patent.

3 44. On information and belief, all of the Accused Instrumentalities perform
4 the claimed methods in substantially the same way, e.g., in the manner specified in the
5 H.264 standard.
6

7 45. On information and belief, use of the Accused Instrumentalities in their
8 ordinary and customary fashion results in infringement of the methods and systems
9 claimed by the '535 patent.
10

11 46. On information and belief, Defendants have had knowledge of the '535
12 patent since at least the filing of this Complaint or shortly thereafter, and on
13 information and belief, Defendants knew of the '535 patent and knew of its
14 infringement, including by way of this lawsuit. By the time of trial, Defendants will
15 have known and intended (since receiving such notice) that its continued actions
16 would actively induce and contribute to the infringement of the claims of the '535
17 patent.
18
19
20

21 47. Upon information and belief, Defendants' affirmative acts of making,
22 using, and selling the Accused Instrumentalities, and providing implementation
23 services and technical support to users of the Accused Instrumentalities, including,
24 e.g., through training, demonstrations, brochures, installation and user guides, have
25 induced and continue to induce users of the Accused Instrumentalities to use them in
26 their normal and customary way to infringe the '535 patent by practicing a method,
27
28

1 comprising: determining a parameter of at least a portion of a data block; selecting one
2 or more asymmetric compressors from among a plurality of compressors based upon
3 the determined parameter or attribute; compressing the at least the portion of the data
4 block with the selected one or more asymmetric compressors to provide one or more
5 compressed data blocks; and storing at least a portion of the one or more compressed
6 data blocks. For example, Defendants adopted H.264 as its video codec in its
7 YouTube TV, YouTube Live Streaming Platform, and YouTube App. As another
8 example, YouTube on its help webpages states that “[A]n **encoder** compresses audio
9 and video into a format that can be delivered to the YouTube platform. YouTube
10 transcodes your content into lower bit rates, including HLS and iOS. This means your
11 content will be available to as wide an audience as possible on desktop, mobile and
12 tablet.”

13
14 *See, e.g.* <https://support.google.com/youtube/answer/6251900>. For similar reasons,
15
16 Defendants also induce their customers to use the Accused Instrumentalities to
17
18 infringe other claims of the ‘535 patent. Defendants specifically intended and was
19
20 aware that these normal and customary activities would infringe the ‘535 patent.

21
22 Defendants performed the acts that constitute induced infringement, and would induce
23
24 actual infringement, with the knowledge of the ‘535 patent and with the knowledge, or
25
26 willful blindness to the probability, that the induced acts would constitute
27
28 infringement. For example, since filing of this action, Defendants know that the
ordinary way of using HTTP Live Streaming—which is directed to choosing different

1 compression techniques based on current or anticipated throughput—in the Accused
2 Instrumentalities infringes the patent but nevertheless continues to promote HTTP
3 Live Streaming to customers. The only reasonable inference is that Defendants
4 specifically intend the users to infringe the patent. On information and belief,
5 Defendants engaged in such inducement to promote the sales of the Accused
6 Instrumentalities. Accordingly, Defendants have induced and continue to induce users
7 of the Accused Instrumentalities to use the Accused Instrumentalities in their ordinary
8 and customary way to infringe the ‘535 patent, knowing that such use constitutes
9 infringement of the ‘535 patent. Accordingly, Defendants have been (as of filing of
10 the original complaint), and currently is, inducing infringement of the ‘535 patent, in
11 violation of 35 U.S.C. § 271(b).

12
13
14
15 48. Defendants have also infringed, and continues to infringe, claims of the
16 ‘535 patent by offering to commercially distribute, commercially distributing, making,
17 and/or importing the Accused Instrumentalities, which are used in practicing the
18 process, or using the systems, of the ‘535 patent, and constitute a material part of the
19 invention. Defendants know the components in the Accused Instrumentalities to be
20 especially made or especially adapted for use in infringement of the ‘535 patent, not a
21 staple article, and not a commodity of commerce suitable for substantial noninfringing
22 use. For example, the ordinary way of using HTTP Live Streaming—which is
23 directed to choosing different compression techniques based on current or anticipated
24 throughput—infringes the patent, and as such, is especially adapted for use in
25
26
27
28

1 infringement. Moreover, there is no substantial noninfringing use, as HTTP Live
2 Streaming is directed to choosing different compression techniques based on current
3 or anticipated throughput. Accordingly, Defendants have been (as of filing of the
4 original complaint), and currently is, contributorily infringing the '535 patent, in
5 violation of 35 U.S.C. § 271(c).
6

7 49. By making, using, offering for sale, selling and/or importing into the
8 United States the Accused Instrumentalities, and touting the benefits of using the
9 Accused Instrumentalities' compression features, Defendants have injured Realtime
10 and is liable to Realtime for infringement of the '535 patent pursuant to 35 U.S.C. §
11 271.
12
13

14 50. As a result of Defendants' infringement of the '535 patent, Plaintiff
15 Realtime is entitled to monetary damages in an amount adequate to compensate for
16 Defendants' infringement, but in no event less than a reasonable royalty for the use
17 made of the invention by Defendants, together with interest and costs as fixed by the
18 Court.
19
20

21 **COUNT III**

22 **INFRINGEMENT OF U.S. PATENT NO. 9,769,477**

23 51. Plaintiff re-alleges and incorporates by reference the foregoing
24 paragraphs, as if fully set forth herein.
25

26 52. On information and belief, Defendants have made, used, offered for sale,
27 sold and/or imported into the United States products that infringe the '477 patent, and
28

1 continues to do so. By way of illustrative example, these infringing products include,
2 without limitation, YouTube's streaming products/services such as, e.g., YouTube
3 site, YouTube TV, YouTube Live Streaming Platform, YouTube App, etc., and all
4 versions and variations thereof since the issuance of the '477 patent ("Accused
5 Instrumentalities").
6

7 53. On information and belief, Defendants have directly infringed and
8 continues to infringe the '477 patent, for example, through its sale, offer for sale,
9 importation, use and testing of the Accused Instrumentalities that practice Claim 1 of
10 the '477 patent, namely, a system, comprising: a plurality of different asymmetric data
11 compression encoders, wherein each asymmetric data compression encoder of the
12 plurality of different asymmetric data compression encoders is configured to utilize
13 one or more data compression algorithms, and wherein a first asymmetric data
14 compression encoder of the plurality of different asymmetric data compression
15 encoders is configured to compress data blocks containing video or image data at a
16 higher data compression rate than a second asymmetric data compression encoder of
17 the plurality of different asymmetric data compression encoders; and one or more
18 processors configured to: determine one or more data parameters, at least one of the
19 determined one or more data parameters relating to a throughput of a communications
20 channel measured in bits per second; and select one or more asymmetric data
21 compression encoders from among the plurality of different asymmetric data
22 compression encoders based upon, at least in part, the determined one or more data
23
24
25
26
27
28

1 parameters.

2 54. For example, the Accused Instrumentalities utilize H.264 video
3 compression standard in delivering live video via HTTP Live Streaming (HLS)
4 technology. For example, YouTube “transcodes your content into lower bit rates,
5 including HLS and iOS.” *See, e.g.,*

6 <https://support.google.com/youtube/answer/6251900>. Furthermore, according to
7 HLS “protocol specification does not limit the encoder selection. However, the current
8 Apple implementation should interoperate with encoders that produce MPEG-2
9 Transport Streams containing H.264 video and AAC audio (HE-AAC or AAC-LC).”
10 *See, e.g.,*

11 [https://developer.apple.com/library/content/documentation/NetworkingInternet/Conc](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html)
12 eptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.
13 html. As another example, HLS developer guide also states: “HTTP Live Streaming
14 supports switching between streams dynamically if the available bandwidth changes.
15 The client software uses heuristics to determine appropriate times to switch between
16 the alternates. Currently, these heuristics are based on recent trends in measured
17 network throughput.” *See, e.g.,*

18 [https://developer.apple.com/library/content/documentation/NetworkingInternet/Conc](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/UsingHTTPLiveStreaming/UsingHTTPLiveStreaming.html)
19 eptual/StreamingMediaGuide/UsingHTTPLiveStreaming/UsingHTTPLiveStreaming.
20 html. Moreover, HLS developer guide states: “The current implementation of the
21 client observes the effective bandwidth while playing a stream. If a higher-quality
22

1 stream is available and the bandwidth appears sufficient to support it, the client
2 switches to a higher quality. If a lower-quality stream is available and the current
3 bandwidth appears insufficient to support the current stream, the client switches to a
4 lower quality.” *See,*

6 *e.g.*, [https://developer.apple.com/library/content/documentation/NetworkingInternet/
7 Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuest
8 ions.html](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html).

10 55. The Accused Instrumentalities include a plurality of different asymmetric
11 data compression encoders, wherein each asymmetric data compression encoder of the
12 plurality of different asymmetric data compression encoders is configured to utilize
13 one or more data compression algorithms, and wherein a first asymmetric data
14 compression encoder of the plurality of different asymmetric data compression
15 encoders is configured to compress data blocks containing video or image data at a
16 higher data compression rate than a second asymmetric data compression encoder of
17 the plurality of different asymmetric data compression encoders. H.264 provides for
18 multiple different ranges of parameters (e.g., bitrate, max video bitrate, resolution
19 parameters, etc.), each included in the “profiles” and “levels” defined by the H.264
20 standard. *See* http://www.axis.com/files/whitepaper/wp_h264_31669_en_0803_lo.pdf
21 at 5:

4. H.264 profiles and levels

The joint group involved in defining H.264 focused on creating a simple and clean solution, limiting options and features to a minimum. An important aspect of the standard, as with other video standards, is providing the capabilities in profiles (sets of algorithmic features) and levels (performance classes) that optimally support popular productions and common formats.

H.264 has seven profiles, each targeting a specific class of applications. Each profile defines what feature set the encoder may use and limits the decoder implementation complexity.

Network cameras and video encoders will most likely use a profile called the baseline profile, which is intended primarily for applications with limited computing resources. The baseline profile is the most suitable given the available performance in a real-time encoder that is embedded in a network video product. The profile also enables low latency, which is an important requirement of surveillance video and also particularly important in enabling real-time, pan/tilt/zoom (PTZ) control in PTZ network cameras.

H.264 has 11 levels or degree of capability to limit performance, bandwidth and memory requirements. Each level defines the bit rate and the encoding rate in macroblock per second for resolutions ranging from QCIF to HDTV and beyond. The higher the resolution, the higher the level required.

56. See https://en.wikipedia.org/wiki/H.264/MPEG-4_AVC:

Levels with maximum property values

Level	Max decoding speed		Max frame size		Max video bit rate for video coding layer (VCL) kbit/s			Examples for high resolution @ highest frame rate (max stored frames) Toggle additional details
	Luma samples/s	Macroblocks/s	Luma samples	Macroblocks	Baseline, Extended and Main Profiles	High Profile	High 10 Profile	
1	380,160	1,485	25,344	99	64	80	192	176x144@15.0 (4)
1b	380,160	1,485	25,344	99	128	160	384	176x144@15.0 (4)
1.1	768,000	3,000	101,376	396	192	240	576	352x288@7.5 (2)
1.2	1,536,000	6,000	101,376	396	384	480	1,152	352x288@15.2 (6)
1.3	3,041,280	11,880	101,376	396	768	960	2,304	352x288@30.0 (6)
2	3,041,280	11,880	101,376	396	2,000	2,500	6,000	352x288@30.0 (6)
2.1	5,068,800	19,800	202,752	792	4,000	5,000	12,000	352x576@25.0 (6)
2.2	5,184,000	20,250	414,720	1,620	4,000	5,000	12,000	720x576@12.5 (5)
3	10,368,000	40,500	414,720	1,620	10,000	12,500	30,000	720x576@25.0 (5)
3.1	27,648,000	108,000	921,600	3,600	14,000	17,500	42,000	1,280x720@30.0 (5)
3.2	55,296,000	216,000	1,310,720	5,120	20,000	25,000	60,000	1,280x1,024@42.2 (4)
4	62,914,560	245,760	2,097,152	8,192	20,000	25,000	60,000	2,048x1,024@30.0 (4)
4.1	62,914,560	245,760	2,097,152	8,192	50,000	62,500	150,000	2,048x1,024@30.0 (4)
4.2	133,693,440	522,240	2,228,224	8,704	50,000	62,500	150,000	2,048x1,080@60.0 (4)
5	150,994,944	589,824	5,652,480	22,080	135,000	168,750	405,000	3,672x1,536@26.7 (5)
5.1	251,658,240	983,040	9,437,184	36,864	240,000	300,000	720,000	4,096x2,304@26.7 (5)
5.2	530,841,600	2,073,600	9,437,184	36,864	240,000	300,000	720,000	4,096x2,304@56.3 (5)

57. A video data block is organized by the group of pictures (GOP) structure, which is a “collection of successive pictures within a coded video stream.” See https://en.wikipedia.org/wiki/Group_of_pictures. A GOP structure can contain intra coded pictures (I picture or I frame), predictive coded pictures (P picture or P frame), bipredictive coded pictures (B picture or B frame) and direct coded pictures (D picture

or D frames, or DC direct coded pictures which are used only in MPEG-1 video). See https://en.wikipedia.org/wiki/Video_compression_picture_types (for descriptions of I frames, P frames and B frames); <https://en.wikipedia.org/wiki/MPEG-1#D-frames> (for descriptions of D frames). Thus, at least a portion of a video data block would also make up a GOP structure and could also contain I frames, P frames, B frames and/or D frames. The GOP structure also reflects the size of a video data block, and the GOP structure can be controlled and used to fine-tune other parameters (e.g. bitrate, max video bitrate and resolution parameters) or even be considered as a parameter by itself.

58. Moreover, compression algorithms utilized in H.264 standard are asymmetric.

Network and Complex Systems
ISSN 2224-610X (Paper) ISSN 2225-0603 (Online)
Vol.4, No.2, 2014

www.iiste.org

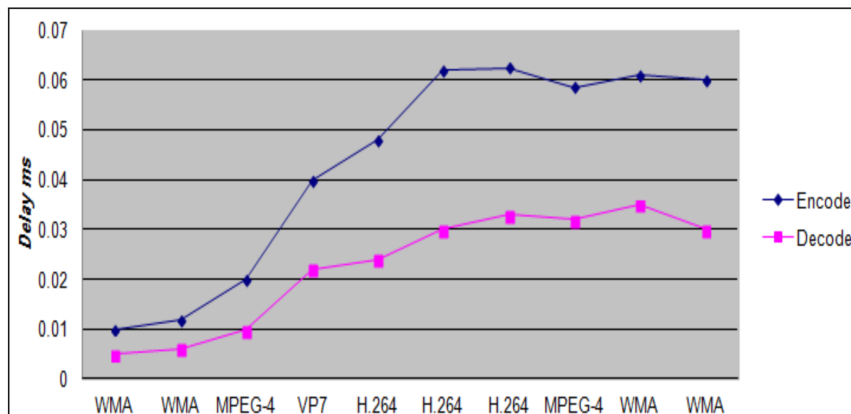


Figure 4: Encoder and Decoder Delay for Variety of Video Codec's

It is clear from figure 4, decoder delay in the most of video codec's can be assumed half of the encode delay.

See e.g., <http://www.iiste.org/Journals/index.php/NCS/article/viewFile/11072/11373>

59. The Accused Instrumentalities include one or more processors configured to: determine one or more data parameters, at least one of the determined one or more data parameters relating to a throughput of a communications channel measured in

1 bits per second; and select one or more asymmetric data compression encoders from
2 among the plurality of different asymmetric data compression encoders based upon, at
3 least in part, the determined one or more data parameters. For example, YouTube
4 “automatically detects the stream resolution and frame rate. When you are live, we’ll
5 transcode to lower resolutions so all of your fans can enjoy your stream no matter the
6 quality of their Internet connection.” *See, e.g.*

7
8
9 https://support.google.com/youtube/answer/2853700?hl=en&ref_topic=6136989

10 Moreover, based on the bitrate and/or resolution parameter identified (e.g. bitrate, max
11 video bitrate, resolution, GOP structure or frame type within a GOP structure), any
12 H.264-compliant system such as the Accused Instrumentalities would determine
13 which profile (e.g., “baseline,” “extended,” “main”, or “high”) corresponds with that
14 parameter, then select between at least two asymmetric compressors. If baseline or
15 extended is the corresponding profile, then the system will select a Context-Adaptive
16 Variable Length Coding (“CAVLC”) entropy encoder. If main or high is the
17 corresponding profile, then the system will select a Context-Adaptive Binary
18 Arithmetic Coding (“CABAC”) entropy encoder. Both encoders are asymmetric
19 compressors because it takes a longer period of time for them to compress data than to
20 decompress data.
21
22
23
24

25 See e.g., <http://www.iiste.org/Journals/index.php/NCS/article/viewFile/11072/11373>
26
27
28

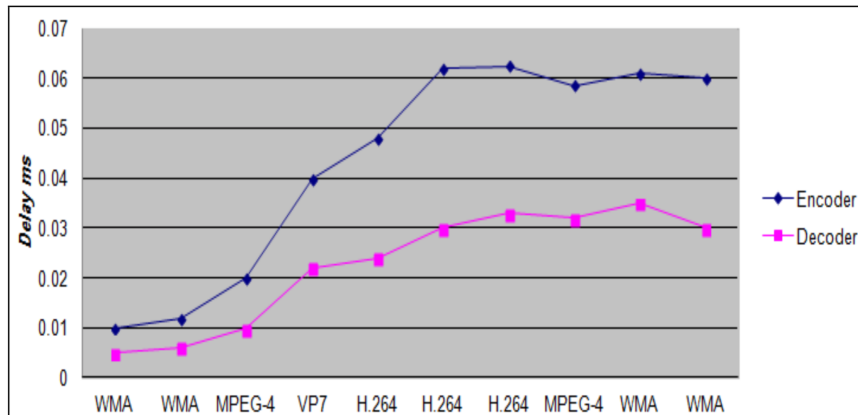


Figure 4: Encoder and Decoder Delay for Variety of Video Codec's

It is clear from figure 4, decoder delay in the most of video codec's can be assumed half of the encode delay.

See <https://sonnati.wordpress.com/2007/10/29/how-h-264-works-part-ii/>

	Baseline	Extended	Main	High	High 10
I and P Slices	Yes	Yes	Yes	Yes	Yes
B Slices	No	Yes	Yes	Yes	Yes
SI and SP Slices	No	Yes	No	No	No
Multiple Reference Frames	Yes	Yes	Yes	Yes	Yes
In-Loop Deblocking Filter	Yes	Yes	Yes	Yes	Yes
CAVLC Entropy Coding	Yes	Yes	Yes	Yes	Yes
CABAC Entropy Coding	No	No	Yes	Yes	Yes
Flexible Macroblock Ordering (FMO)	Yes	Yes	No	No	No
Arbitrary Slice Ordering (ASO)	Yes	Yes	No	No	No
Redundant Slices (RS)	Yes	Yes	No	No	No
Data Partitioning	No	Yes	No	No	No
Interlaced Coding (PicAFF, MBAFF)	No	Yes	Yes	Yes	Yes
4:2:0 Chroma Format	Yes	Yes	Yes	Yes	Yes
Monochrome Video Format (4:0:0)	No	No	No	Yes	Yes
4:2:2 Chroma Format	No	No	No	No	No
4:4:4 Chroma Format	No	No	No	No	No
8 Bit Sample Depth	Yes	Yes	Yes	Yes	Yes
9 and 10 Bit Sample Depth	No	No	No	No	Yes
11 to 14 Bit Sample Depth	No	No	No	No	No
8x8 vs. 4x4 Transform Adaptivity	No	No	No	Yes	Yes
Quantization Scaling Matrices	No	No	No	Yes	Yes
Separate Cb and Cr QP control	No	No	No	Yes	Yes
Separate Color Plane Coding	No	No	No	No	No
Predictive Lossless Coding	No	No	No	No	No

See http://web.cs.ucla.edu/classes/fall03/cs218/paper/H.264_MPEG4_Tutorial.pdf at 7:

The following table summarizes the two major types of entropy coding: Variable Length Coding (VLC) and Context Adaptive Binary Arithmetic Coding (CABAC). CABAC offers superior coding efficiency over VLC by adapting to the changing probability distribution of symbols, by exploiting correlation between symbols, and by adaptively exploiting bit correlations using arithmetic coding. H.264 also supports Context Adaptive Variable Length Coding (CAVLC) which offers superior entropy coding over VLC without the full cost of CABAC.

H.264 Entropy Coding – Comparison of Approaches

Characteristics	Variable Length Coding (VLC)	Context Adaptive Binary Arithmetic Coding(CABAC)
• Where it is used	MPEG-2, MPEG-4 ASP	H.264/MPEG-4 AVC (high efficiency option)
• Probability distribution	Static - Probabilities never change	Adaptive - Adjusts probabilities based on actual data
• Leverages correlation between symbols	No - Conditional probabilities ignored	Yes - Exploits symbol correlations by using "contexts"
• Non-integer code words	No - Low coding efficiency for high probability symbols	Yes - Exploits "arithmetic coding" which generates non-integer code words for higher efficiency

60. Moreover, the H.264 Standard requires a bit-flag descriptor, which is set to determine the correct decoder for the corresponding encoder. As shown below, if the flag = 0, then CAVLC must have been selected as the encoder; if the flag = 1, then CABAC must have been selected as the encoder. See https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-H.264-201304-S!!PDF-E&type=items (Rec. ITU-T H.264 (04/2013)) at 80:

entropy_coding_mode_flag selects the entropy decoding method to be applied for the syntax elements for which two descriptors appear in the syntax tables as follows:

- If **entropy_coding_mode_flag** is equal to 0, the method specified by the left descriptor in the syntax table is applied (Exp-Golomb coded, see clause 9.1 or CAVLC, see clause 9.2).
- Otherwise (**entropy_coding_mode_flag** is equal to 1), the method specified by the right descriptor in the syntax table is applied (CABAC, see clause 9.3).

61. The processor in the Accused Instrumentalities decides which compression (e.g., CABAC, CAVLC, etc.) to use at a point in time based on parameters, for example, e.g., current or anticipated throughput. For example, when a low bandwidth is present, the Accused Instrumentalities select lower quality stream using a particular compression technique. As another example, when a high

bandwidth is present, the Accused Instrumentalities select higher quality stream using another particular compression technique. For example, the Accused Instrumentalities' use of HTTP Live Streaming is directed to this selection. As another example, the Accused Instrumentalities' use of different "Profiles" of H.264 is directed to selecting lower quality stream using a particular compression technique (e.g., CABAC or CAVLC, etc.) for lower anticipated bandwidth situations, and selecting higher quality stream using a higher compression technique (e.g., CABAC or CAVLC, etc.) for higher anticipated bandwidth situations.

62. After its selection, the asymmetric compressor (CAVLC or CABAC) will compress the video data to provide various compressed data blocks, which can be organized in a GOP structure (see above). *See*

<https://sonnati.wordpress.com/2007/10/29/how-h-264-works-part-ii/>:

Entropy Coding

For entropy coding, H.264 may use an enhanced VLC, a more complex context-adaptive variable-length coding (CAVLC) or an ever more complex Context-adaptive binary-arithmetic coding (CABAC) which are complex techniques to losslessly compress syntax elements in the video stream knowing the probabilities of syntax elements in a given context. The use of CABAC can improve the compression of around 5-7%. CABAC may requires a 30-40% of total processing power to be accomplished.

See

<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.602.1581&rep=rep1&type=pdf> at 13:

Typical compression ratios to maintain excellent quality are:

- 10:1 for general images using JPEG
- 30:1 for general video using H.263 and MPEG-2
- 60:1 for general video using H.264 and WMV9

See http://www.ijera.com/papers/Vol3_issue4/BM34399403.pdf at 2:

1 Most visual communication systems today
2 use Baseline Profile. Baseline is the simplest H.264
3 profile and defines, for example, zigzag scanning of
4 the picture and using 4:2:0 (YUV video formats)
5 chrominance sampling. In Baseline Profile, the
6 picture is split in blocks consisting of 4x4 pixels,
7 and each block is processed separately. Another
8 important element of the Baseline Profile is the use
9 of Universal Variable Length Coding (UVLC) and
10 Context Adaptive Variable Length Coding
11 (CAVLC) entropy coding techniques.

12 The Extended and Main Profiles includes
13 the functionality of the Baseline Profile and add
14 improvements to the predictions algorithms. Since
15 transmitting every single frame (think 30 frames per
16 second for good quality video) is not feasible if you
17 are trying to reduce the bit rate 1000-2000 times,
18 temporal and motion prediction are heavily used in
19 H.264, and allow transmitting only the difference
20 between one frame and the previous frames. The
21 result is spectacular efficiency gain, especially for
22 scenes with little change and motion.

23 The High Profile is the most powerful
24 profile in H.264, and it allows most efficient coding
25 of video. For example, large coding gain achieved
26 through the use of Context Adaptive Binary
27 Arithmetic Coding (CABAC) encoding which is
28 more efficient than the UVLC/CAVLC used in
Baseline Profile.

The High Profile also uses adaptive
transform that decides on the fly if 4x4 or 8x8-pixel
blocks should be used. For example, 4x4 blocks are
used for the parts of the picture that are dense with
detail, while parts that have little detail are
transformed using 8x8 blocks.

63. On information and belief, Defendants also directly infringe and continue
to infringe other claims of the '477 patent.

64. On information and belief, all of the Accused Instrumentalities perform
the claimed methods in substantially the same way, e.g., in the manner specified in the
H.264 standard.

65. On information and belief, use of the Accused Instrumentalities in their
ordinary and customary fashion results in infringement of the methods and systems
claimed by the '477 patent.

66. On information and belief, Defendants have had knowledge of the '477

1 patent since at least the filing of this Complaint or shortly thereafter, and on
2 information and belief, Defendants knew of the '477 patent and knew of its
3 infringement, including by way of this lawsuit. By the time of trial, Defendants will
4 have known and intended (since receiving such notice) that its continued actions
5 would actively induce and contribute to the infringement of the claims of the '477
6 patent.
7

8
9 67. Upon information and belief, Defendants' affirmative acts of making,
10 using, and selling the Accused Instrumentalities, and providing implementation
11 services and technical support to users of the Accused Instrumentalities, including,
12 e.g., through training, demonstrations, brochures, installation and user guides, have
13 induced and continue to induce users of the Accused Instrumentalities to use them in
14 their normal and customary way to infringe the '477 patent by using a system
15 comprising: a plurality of different asymmetric data compression encoders, wherein
16 each asymmetric data compression encoder of the plurality of different asymmetric
17 data compression encoders is configured to utilize one or more data compression
18 algorithms, and wherein a first asymmetric data compression encoder of the plurality
19 of different asymmetric data compression encoders is configured to compress data
20 blocks containing video or image data at a higher data compression rate than a second
21 asymmetric data compression encoder of the plurality of different asymmetric data
22 compression encoders; and one or more processors configured to: determine one or
23 more data parameters, at least one of the determined one or more data parameters
24
25
26
27
28

1 relating to a throughput of a communications channel measured in bits per second;
2 and select one or more asymmetric data compression encoders from among the
3 plurality of different asymmetric data compression encoders based upon, at least in
4 part, the determined one or more data parameters. For example, Defendants adopted
5 H.264 as its video codec in its YouTube TV, YouTube Live Streaming Platform, and
6 YouTube App. As another example, YouTube on its help webpages states that
7 “[A]n **encoder** compresses audio and video into a format that can be delivered to the
8 YouTube platform. YouTube transcodes your content into lower bit rates, including
9 HLS and iOS. This means your content will be available to as wide an audience as
10 possible on desktop, mobile and tablet.” *See, e.g.*
11 <https://support.google.com/youtube/answer/6251900>. For similar reasons, Defendants
12 also induce their customers to use the Accused Instrumentalities to infringe other
13 claims of the ’477 patent. Defendants specifically intended and was aware that these
14 normal and customary activities would infringe the ’477 patent. Defendants
15 performed the acts that constitute induced infringement, and would induce actual
16 infringement, with the knowledge of the ’477 patent and with the knowledge, or
17 willful blindness to the probability, that the induced acts would constitute
18 infringement. For example, since filing of this action, Defendants know that the
19 ordinary way of using HTTP Live Streaming—which is directed to choosing different
20 compression techniques based on current or anticipated throughput—in the Accused
21 Instrumentalities infringes the patent but nevertheless continues to promote HTTP
22
23
24
25
26
27
28

1 Live Streaming to customers. The only reasonable inference is that Defendants
2 specifically intend the users to infringe the patent. On information and belief,
3 Defendants engaged in such inducement to promote the sales of the Accused
4 Instrumentalities. Accordingly, Defendants have induced and continue to induce users
5 of the Accused Instrumentalities to use the Accused Instrumentalities in their ordinary
6 and customary way to infringe the '477 patent, knowing that such use constitutes
7 infringement of the '477 patent. Accordingly, Defendants have been (as of filing of
8 the original complaint), and currently are, inducing infringement of the '477 patent, in
9 violation of 35 U.S.C. § 271(b).
10
11
12

13 68. Defendants have also infringed, and continues to infringe, claims of
14 the '477 patent by offering to commercially distribute, commercially distributing,
15 making, and/or importing the Accused Instrumentalities, which are used in practicing
16 the process, or using the systems, of the '477 patent, and constitute a material part of
17 the invention. Defendants know the components in the Accused Instrumentalities to
18 be especially made or especially adapted for use in infringement of the '477 patent,
19 not a staple article, and not a commodity of commerce suitable for substantial
20 noninfringing use. For example, the ordinary way of using HTTP Live Streaming—
21 which is directed to choosing different compression techniques based on current or
22 anticipated throughput—infringes the patent, and as such, is especially adapted for use
23 in infringement. Moreover, there is no substantial noninfringing use, as HTTP Live
24 Streaming is directed to choosing different compression techniques based on current
25
26
27
28

1 or anticipated throughput. Accordingly, Defendants have been (as of filing of the
2 original complaint), and currently is, contributorily infringing the '477 patent, in
3 violation of 35 U.S.C. § 271(c).
4

5 69. By making, using, offering for sale, selling and/or importing into the
6 United States the Accused Instrumentalities, and touting the benefits of using the
7 Accused Instrumentalities' compression features, Defendants have injured Realtime
8 and is liable to Realtime for infringement of the '477 patent pursuant to 35 U.S.C. §
9 271.
10

11 70. As a result of Defendants' infringement of the '477 patent, Plaintiff
12 Realtime is entitled to monetary damages in an amount adequate to compensate for
13 Defendants' infringement, but in no event less than a reasonable royalty for the use
14 made of the invention by Defendants, together with interest and costs as fixed by the
15 Court.
16
17

18 **COUNT IV**

19 **INFRINGEMENT OF U.S. PATENT NO. 8,634,462**

20
21 71. Plaintiff re-alleges and incorporates by reference the foregoing
22 paragraphs, as if fully set forth herein.
23

24 72. On information and belief, Defendants have made, used, offered for sale,
25 sold and/or imported into the United States products that infringe the '462 patent, and
26 continues to do so. By way of illustrative example, these infringing products include,
27 without limitation, Defendants' products and services that implement the High
28

1 Efficiency Video Coding (HEVC; also known as H.265) standard (e.g., YouTube,
2 Google Photos, Chromecast Ultra, Google Duo, etc.), and all versions and variations
3 thereof since the issuance of the '462 patent ("Accused Instrumentalities").
4

5 73. On information and belief, Defendants have directly infringed and
6 continues to infringe the '462 patent, for example, through its sale, offer for sale,
7 importation, use and testing of the Accused Instrumentalities, which practices the
8 method claimed by Claim 1 of the '462 patent, namely, a method for coding a video
9 signal using hybrid coding, comprising: reducing temporal redundancy by block based
10 motion compensated prediction in order to establish a prediction error signal;
11 performing quantization on samples of the prediction error signal or on coefficients
12 resulting from a transformation of the prediction error signal into the frequency
13 domain to obtain quantized values, representing quantized samples or quantized
14 coefficients respectively, wherein the prediction error signal includes a plurality of
15 subblocks each including a plurality of quantized values; calculating a first
16 quantization efficiency for the quantized values of at least one subblock of the
17 plurality of subblocks; setting the quantized values of the at least one subblock to all
18 zeroes; calculating a second quantization efficiency for the at least one subblock while
19 all of the quantized values are zeroes; selecting which of the first and second
20 quantization efficiencies is a higher efficiency; and selecting, for further proceeding,
21 the at least one subblock with the quantized values prior to setting the quantized
22 values of the at least one subblock to all zeroes if the first quantization efficiency is
23
24
25
26
27
28

1 higher and selecting the at least one subblock with the quantized values set to zero, for
2 further proceeding, if the second quantization efficiency is higher. Upon information
3 and belief, Defendants use the Accused Instrumentalities to practice infringing
4 methods for its own internal non-testing business purposes, while testing the Accused
5 Instrumentalities, and while providing technical support and repair services for the
6 Accused Instrumentalities to Defendants' customers.
7

8
9 74. For example, the Accused Instrumentalities utilize the HEVC standard.
10 *See, e.g.*, [https://support.google.com/youtube/troubleshooter/2888402?hl=en&vid=0-](https://support.google.com/youtube/troubleshooter/2888402?hl=en&vid=0-315731740723-1524058922197)
11 [315731740723-1524058922197](https://support.google.com/youtube/troubleshooter/2888402?hl=en&vid=0-315731740723-1524058922197) ("Supported YouTube file formats ... HEVC
12 (h265)"); <https://plus.google.com/+PeggyKTC/posts/VTD6DM5Rxvx> ("Google
13 Photos now lets iOS11 users back up HEIF photos and HEVC videos");
14 <https://developers.google.com/cast/docs/media#image-formats> ("Supported Media for
15 Google Cast ... Video codecs (Chromecast Ultra) ... HEVC / H.265");
16 <https://wccftech.com/google-duo-v26-brings-support-h-265/> ("Google Duo V26
17 Brings Support for H.265").
18
19
20

21 75. The HEVC Specification (e.g., ITU-T H.265 Series H: Audiovisual and
22 Multimedia Systems, "Infrastructure of audiovisual services – Coding of moving
23 video" High efficiency video coding) ("HEVC Spec") sets forth standard that is
24 followed by HEVC compliant devices, and is relevant to both decoding and encoding
25 that are performed pursuant to the HEVC standard.
26

27 76. The Accused Instrumentalities performs a method for coding a video
28

1 signal using hybrid coding. For instance, Accused Instrumentalities performs a
2 method for coding a video signal using hybrid coding when performing coding using
3 HEVC. For example, the aim of the coding process is the production of a bitstream,
4 as defined in definition 3.12 of the ITU-T H.265 Series H: Audiovisual and
5 Multimedia Systems, “Infrastructure of audiovisual services – Coding of moving
6 video” High efficiency video coding (“HEVC Spec”): “bitstream: A sequence of bits,
7 in the form of a NAL unit stream or a byte stream, that forms the representation of
8 coded pictures and associated data forming one or more coded video sequences
9 (CVSs).” *See also, e.g.*, “Overview of the High Efficiency Video Coding (HEVC)
10 Standard” by Gary J. Sullivan, Fellow, IEEE, Jens-Rainer Ohm, Member, IEEE,
11 Woo-Jin Han, Member, IEEE, and Thomas Wiegand, Fellow, IEEE, published in
12 IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO
13 TECHNOLOGY, VOL. 22, NO. 12, DECEMBER 2012 (“IEEE HEVC”) (“The video
14 coding layer of HEVC employs the same hybrid approach (inter-/intrapicture
15 prediction and 2-D transform coding) used in all video compression standards since
16 H.261”). *See also, e.g.*, HEVC Spec at 0.7 “Overview of the design characteristics.”

22 77. The Accused Instrumentalities reduce temporal redundancy by block
23 based motion compensated prediction in order to establish a prediction error signal.
24 For instance, the Accused Instrumentalities reduce temporal redundancy by block
25 based motion compensated prediction in order to establish a prediction error signal
26 when performing HEVC encoding. For example, clause 8.5.3 Decoding process for
27
28

1 prediction units in inter prediction mode and the subclauses thereof of the HEVC Spec
2 describe the block based motion compensation techniques used in the decoding
3 process, which indicate that block based motion compensation prediction is used in
4 encoding to reduce temporal redundancy and establish a prediction error signal. *See*
5 *also, e.g., IEEE HEVC at 1651-1652* 6) Motion compensation: Quarter-sample
6 precision is used for the MVs, and 7-tap or 8-tap filters are used for interpolation of
7 fractional-sample positions (compared to six-tap filtering of half-sample positions
8 followed by linear interpolation for quarter-sample positions in H.264/MPEG-4
9 AVC). Similar to H.264/MPEG-4 AVC, multiple reference pictures are used. For each
10 PB, either one or two motion vectors can be transmitted, resulting either in
11 unipredictive or bipredictive coding, respectively. As in H.264/MPEG-4 AVC, a
12 scaling and offset operation may be applied to the prediction signal(s) in a manner
13 known as weighted prediction.”).

14 78. The Accused Instrumentalities perform quantization on samples of the
15 prediction error signal or on coefficients resulting from a transformation of the
16 prediction error signal into the frequency domain to obtain quantized values,
17 representing quantized samples or quantized coefficients respectively. For instance,
18 the Accused Instrumentalities perform quantization on samples of the prediction error
19 signal or on coefficients resulting from a transformation of the prediction error signal
20 into the frequency domain to obtain quantized values, representing quantized samples
21 or quantized coefficients respectively when performing HEVC coding. For example,
22
23
24
25
26
27
28

1 the quantization parameter and the scaling (inverse quantization) are defined in
2 definitions 3.112 (page 10) and 3.131 (page 11), respectively, the usage of the scaling
3 process in the decoding being described in clause and 8.6 Scaling, transformation and
4 array construction process prior to deblocking filter process of the HEVC Spec, which
5 indicate quantization as claimed when doing HEVC encoding. *See also, e.g., IEEE*
6 *HEVC at 1652 (“8) Quantization control: As in H.264/MPEG-4 AVC, uniform*
7 *reconstruction quantization (URQ) is used in HEVC, with quantization scaling*
8 *matrices supported for the various transform block sizes.”).*

11 79. The Accused Instrumentalities perform a method wherein the prediction
12 error signal includes a plurality of subblocks each including a plurality of quantized
13 values. For instance, the Accused Instrumentalities perform a method wherein the
14 prediction error signal includes a plurality of subblocks each including a plurality of
15 quantized values when performing HEVC encoding. For example, the quantized
16 samples or transform coefficients from the subblock are scaled and transformed as
17 described in above mentioned clause 8.6 of the HEVC Spec, indicating prediction
18 error signal as claimed when doing HEVC encoding. *See also, e.g., IEEE HEVC at*
19 *1652 (“Prediction units and prediction blocks (PBs): The decision whether to code a*
20 *picture area using interpicture or intrapicture prediction is made at the CU level. A PU*
21 *partitioning structure has its root at the CU level. Depending on the basic prediction-*
22 *type decision, the luma and chroma CBs can then be further split in size and predicted*
23 *from luma and chroma prediction blocks (PBs). HEVC supports variable PB sizes*
24
25
26
27
28

1 from 64×64 down to 4×4 samples.”).

2 80. The Accused Instrumentalities perform a method of calculating a first
3 quantization efficiency for the quantized values of at least one subblock of the
4 plurality of subblocks; setting the quantized values of the at least one subblock to all
5 zeroes; calculating a second quantization efficiency for the at least one subblock while
6 all of the quantized values are zeroes; selecting which of the first and second
7 quantization efficiencies is a higher efficiency; and selecting, for further proceeding,
8 the at least one subblock with the quantized values prior to setting the quantized
9 values of the at least one subblock to all zeroes if the first quantization efficiency is
10 higher and selecting the at least one subblock with the quantized values set to zero, for
11 further proceeding, if the second quantization efficiency is higher. For instance, the
12 Accused Instrumentalities perform a method of calculating a first quantization
13 efficiency for the quantized values of at least one subblock of the plurality of
14 subblocks; setting the quantized values of the at least one subblock to all zeroes;
15 calculating a second quantization efficiency for the at least one subblock while all of
16 the quantized values are zeroes; selecting which of the first and second quantization
17 efficiencies is a higher efficiency; and selecting, for further proceeding, the at least
18 one subblock with the quantized values prior to setting the quantized values of the at
19 least one subblock to all zeroes if the first quantization efficiency is higher and
20 selecting the at least one subblock with the quantized values set to zero, for further
21 proceeding, if the second quantization efficiency is higher when performing HEVC
22
23
24
25
26
27
28

1 encoding. For example, the bitstream resulting from the encoding as described in this
 2 last item of the claim contains all the relevant information as needed by the decoder
 3 for proper decoding. If the coefficients of the subblock are set to zero as a
 4 consequence of the efficiency calculation, the coded_sub_block_flag, as described in
 5 clause 7.4.9.11 Residual coding semantics, HEVC Spec, is set to 0, indicating that all
 6 the 16 coefficients of the coded sub block have been set to 0:
 7

8 “coded_sub_block_flag[xS][yS] specifies the following for the sub-block at
 9 location (xS, yS) within the current transform block, where a sub-block is a (4x4)
 10 array of 16 transform coefficient levels: – If coded_sub_block_flag[xS][yS] is equal
 11 to 0, the 16 transform coefficient levels of the sub-block at location (xS, yS) are
 12 inferred to be equal to 0.”
 13

14
 15 81. When coded_sub_block_flag[xS][yS] has not been set equal to 0, the
 16 position in the array of non 0 coefficients can be determined as follows:
 17

18 – Otherwise (coded_sub_block_flag[xS][yS] is equal to 1), the following
 19 applies:
 20

21 – If (xS, yS) is equal to (0, 0) and (LastSignificantCoeffX,
 22 LastSignificantCoeffY) is not equal to (0, 0), at least one of the 16
 23 sig_coeff_flag syntax elements is present for the sub-block at location
 24 (xS, yS).
 25

26 – Otherwise, at least one of the 16 transform coefficient levels of the sub-
 27 block at location (xS, yS) has a non zero value.
 28

When `coded_sub_block_flag[xS][yS]` is not present, it is inferred as follows:

– If one or more of the following conditions are true,

`coded_sub_block_flag[xS][yS]` is inferred to be equal to 1:

– (xS, yS) is equal to $(0, 0)$

– (xS, yS) is equal to $(\text{LastSignificantCoeffX} \gg 2,$

$\text{LastSignificantCoeffY} \gg 2)$

– Otherwise, `coded_sub_block_flag[xS][yS]` is inferred to be equal to 0.

HEVC Spec at 7.4.9.11 Residual coding semantics. Therefore, even though the coding algorithms that can be used for reaching specific efficiency targets are not specified by the HEVC Spec (as stated in clause 0.7), this particular combination of choices produces a valid bitstream that has to be decoded by a conformant decoder.

82. The infringement of the Accused Instrumentalities is also shown by way of considering the reference software (*see, e.g.,* <https://hevc.hhi.fraunhofer.de/>). The Accused Instrumentalities implement HEVC encoding in the same or substantially the same manner as the reference software, i.e., in infringing manner. Setting the flag `RDOQ=true` in the encoder configuration file enables rate-distortion-optimized quantization for transformed TUs. This feature is implemented in the HM reference software as function `xRateDistOptQuant` in file `TComTrQuant.cpp`. In the function `xRateDistOptQuant`, the efficiency for setting all quantized values to zero is calculated and stored in the variable `d64BestCost`. In the variable `iBestLastIdxP1`, a 0 is stored indicating that all values starting from the 0th position are set to zero. Afterwards, the

1 efficiency for keeping quantized values unequal to zero is calculated and stored in the
 2 variable totalCost. The variable iBestLastIdxP1 is adjusted correspondingly to values
 3 unequal to 0. The two efficiencies d64BestCost and totalCost are compared, and
 4 selecting for further proceeding either quantized values, which are all set to zero or
 5 quantized values, which are not all set to zero. All values starting from the position
 6 defined by the variable iBestLastIdxP1 are set to zero.

83. Calculation of the efficiency for setting all quantized values to zero and
 storing the result in the variable d64BestCost:

```

Double d64BestCost = 0;
Int ui16CtxCbf = 0;
Int iBestLastIdxP1 = 0;
if( !pcCU->isIntra( uiAbsPartIdx ) && isLuma(compID) && pcCU->getTransformIdx( uiAbsPartIdx ) == 0 )
{
    ui16CtxCbf = 0;
    d64BestCost = d64BlockUncodedCost + xGetICost( m_pcEstBitsSbac->blockRootCbpBits[ ui16CtxCbf ][ 0 ] );
    d64BaseCost += xGetICost( m_pcEstBitsSbac->blockRootCbpBits[ ui16CtxCbf ][ 1 ] );
}
else
{
    ui16CtxCbf = pcCU->getCtxQtCbf( rTu, channelType );
    ui16CtxCbf += getCBFContextOffset(compID);
    d64BestCost = d64BlockUncodedCost + xGetICost( m_pcEstBitsSbac->blockCbpBits[ ui16CtxCbf ][ 0 ] );
    d64BaseCost += xGetICost( m_pcEstBitsSbac->blockCbpBits[ ui16CtxCbf ][ 1 ] );
}
  
```

HEVC Reference Software (<https://hevc.hhi.fraunhofer.de/>).

84. Calculating the efficiency for keeping quantized values unequal to zero
 and storing the result in the variable totalCost:

```

Bool bFoundLast = false;
for (Int iCGScanPos = iCGLastScanPos; iCGScanPos >= 0; iCGScanPos--)
{
    UInt uiCGBlkPos = codingParameters.scanCG[ iCGScanPos ];

    d64BaseCost -= pdCostCoeffGroupSig [ iCGScanPos ];
    if (uiSigCoeffGroupFlag[ uiCGBlkPos ])
    {
        for (Int iScanPosinCG = uiCGSize-1; iScanPosinCG >= 0; iScanPosinCG--)
        {
            iScanPos = iCGScanPos*uiCGSize + iScanPosinCG;

            if (iScanPos > iLastScanPos) continue;
            UInt uiBlkPos = codingParameters.scan[iScanPos];

            if (piDstCoeff[ uiBlkPos ] )
            {
                UInt uiPosY = uiBlkPos >> uiLog2BlockWidth;
                UInt uiPosX = uiBlkPos - ( uiPosY << uiLog2BlockWidth );

                Double d64CostLast= codingParameters.scanType == SCAN_VER ? xGetRateLast( uiPosY, uiPosX, compID ) :
                                     xGetRateLast( uiPosX, uiPosY, compID );
                Double totalCost = d64BaseCost + d64CostLast - pdCostSig[ iScanPos ];
            }
        }
    }
}
  
```

1 HEVC Reference Software (<https://hevc.hhi.fraunhofer.de/>).

2 85. Comparing the two efficiencies d64BestCost and totalCost:

```
3
4  if( totalCost < d64BestCost )
5  {
6      iBestLastIdxP1 = iScanPos + 1;
7      d64BestCost     = totalCost;
8  }
```

9 HEVC Reference Software (<https://hevc.hhi.fraunhofer.de/>).

10 86. Selecting for further proceeding either quantized values, which are all set
11 to zero or quantized values, which are not all set to zero:

```
12 //===== clean uncoded coefficients =====
13 for ( Int scanPos = iBestLastIdxP1; scanPos <= iLastScanPos; scanPos++ )
14 {
15     piDstCoeff[ codingParameters.scan[ scanPos ] ] = 0;
16 }
```

17 HEVC Reference Software (<https://hevc.hhi.fraunhofer.de/>).

18 87. On information and belief, Defendants also directly infringe and continue
19 to infringe other claims of the '462 patent.

20 88. On information and belief, all of the Accused Instrumentalities perform
21 the claimed methods in substantially the same way, e.g., in the manner specified in the
22 HEVC standard.

23 89. On information and belief, use of the Accused Instrumentalities in their
24 ordinary and customary fashion results in infringement of the methods claimed by the
25 '462 patent.

26 90. On information and belief, Defendants have had knowledge of the '462
27 patent since at least the filing of this Complaint or shortly thereafter, and on
28

1 information and belief, Defendants knew of the '462 patent and knew of its
2 infringement, including by way of this lawsuit. By the time of trial, Defendants will
3 have known and intended (since receiving such notice) that its continued actions
4 would actively induce and contribute to the infringement of the claims of the '462
5 patent.
6

7 91. Upon information and belief, Defendants' affirmative acts of making,
8 using, and selling the Accused Instrumentalities, and providing implementation
9 services and technical support to users of the Accused Instrumentalities, including,
10 e.g., through training, demonstrations, brochures, installation and user guides, have
11 induced and continue to induce users of the Accused Instrumentalities to use them in
12 their normal and customary way to infringe the '462. For example, Defendants
13 adopted HEVC as its video codec in its products and services. For similar reasons,
14 Defendants also induce customers to use the Accused Instrumentalities to infringe
15 other claims of the '462 patent. Defendants specifically intended and was aware that
16 these normal and customary activities would infringe the '462 patent. Defendants
17 performed the acts that constitute induced infringement, and would induce actual
18 infringement, with the knowledge of the '462 patent and with the knowledge, or
19 willful blindness to the probability, that the induced acts would constitute
20 infringement. For example, since filing of this action, Defendants know that the
21 ordinary way of using HEVC in the Accused Instrumentalities infringes the patent but
22 nevertheless continues to promote HEVC to customers. The only reasonable
23
24
25
26
27
28

1 inference is that Defendants specifically intends the users to infringe the patent. On
2 information and belief, Defendants engaged in such inducement to promote the sales
3 of the Accused Instrumentalities. Accordingly, Defendants have induced and continue
4 to induce users of the Accused Instrumentalities to use the Accused Instrumentalities
5 in their ordinary and customary way to infringe the '462 patent, knowing that such use
6 constitutes infringement of the '462 patent. Accordingly, Defendants have been (as of
7 filing of the original complaint), and currently are, inducing infringement of the '462
8 patent, in violation of 35 U.S.C. § 271(b).

11 92. Defendants have also infringed, and continues to infringe, claims of the
12 '462 patent by offering to commercially distribute, commercially distributing, making,
13 and/or importing the Accused Instrumentalities, which are used in practicing the
14 process, or using the systems, of the '462 patent, and constitute a material part of the
15 invention. Defendants knows the components in the Accused Instrumentalities to be
16 especially made or especially adapted for use in infringement of the '462 patent, not a
17 staple article, and not a commodity of commerce suitable for substantial noninfringing
18 use. For example, the ordinary way of using HEVC infringes the patent, and as such,
19 is especially adapted for use in infringement with no substantial noninfringing use.
20 Accordingly, Defendants have been (as of filing of the original complaint), and
21 currently are, contributorily infringing the '462 patent, in violation of 35 U.S.C. §
22 271(c).

27 93. By making, using, offering for sale, selling and/or importing into the
28

1 United States the Accused Instrumentalities, and touting the benefits of using the
2 Accused Instrumentalities' compression features, Defendants have injured Realtime
3 and is liable to Realtime for infringement of the '462 patent pursuant to 35 U.S.C. §
4 271.
5

6 94. As a result of Defendants' infringement of the '462 patent, Plaintiff
7 Realtime is entitled to monetary damages in an amount adequate to compensate for
8 Defendants' infringement, but in no event less than a reasonable royalty for the use
9 made of the invention by Defendants, together with interest and costs as fixed by the
10 Court.
11

12 **COUNT V**

13 **INFRINGEMENT OF U.S. PATENT NO. 9,578,298**

14 95. Plaintiff re-alleges and incorporates by reference the foregoing
15 paragraphs, as if fully set forth herein.
16

17 96. On information and belief, Defendants have made, used, offered for sale,
18 sold and/or imported into the United States products that infringe the '298 patent, and
19 continues to do so. By way of illustrative example, these infringing products include,
20 without limitation, Defendants' products and services that implement the High
21 Efficiency Video Coding (HEVC; also known as H.265) standard (YouTube, Google
22 Photos, Chromecast Ultra, Google Duo, Android 5.0+, etc.), and all versions and
23 variations thereof since the issuance of the '298 patent ("Accused Instrumentalities").
24
25
26
27

28 97. On information and belief, Defendants have directly infringed and

1 continues to infringe the '298 patent, for example, through its sale, offer for sale,
2 importation, use and testing of the Accused Instrumentalities, which practices the
3 method claimed by Claim 1 of the '298 patent, namely, a method for processing a
4 video stream of digital images, the method comprising the steps of: receiving the
5 video stream which comprises at least one composite frame (FC), each composite
6 frame containing a pair of stereoscopic digital images (L,R) according to a
7 predetermined frame packing format; generating an output video stream which can be
8 reproduced on a visualization apparatus, receiving metadata which determine an area
9 occupied by one of the two images within said composite frame (FC), said metadata
10 indicating either a geometry of the frame packing format or a frame packing type of
11 said composite frame (FC); determining the area in the composite frame (FC) which is
12 occupied by said one image of the stereoscopic pair within the composite frame based
13 on said metadata; decoding only that part of the composite frame (FC) which contains
14 said one image to be displayed, and generating an output frame containing said
15 decoded image. Upon information and belief, Defendants use the Accused
16 Instrumentalities to practice infringing methods for its own internal non-testing
17 business purposes, while testing the Accused Instrumentalities, and while providing
18 technical support and repair services for the Accused Instrumentalities to Defendants'
19 customers.

20
21
22
23
24
25
26 98. For example, the Accused Instrumentalities utilize the HEVC standard.
27 *See, e.g.,* <https://support.google.com/youtube/troubleshooter/2888402?hl=en&vid=0->
28

1 315731740723-1524058922197 (“Supported YouTube file formats ... HEVC
 2 (h265”); <https://plus.google.com/+PeggyKTC/posts/VTD6DM5Rxvx> (“Google
 3 Photos now lets iOS11 users back up HEIF photos and HEVC videos”);
 4 <https://developers.google.com/cast/docs/media#image-formats> (“Supported Media for
 5 Google Cast ... Video codecs (Chromecast Ultra) ... HEVC / H.265”);
 6 <https://wccfttech.com/google-duo-v26-brings-support-h-265/> (“Google Duo V26
 7 Brings Support for H.265”); [https://developer.android.com/guide/topics/media/media-
 8 formats.html](https://developer.android.com/guide/topics/media/media-formats.html) (“Video formats and codecs ... H.265 HEVC ... (Android 5.0+)”).

11 99. The Accused Instrumentalities receive the video stream which comprises
 12 at least one composite frame (FC), each composite frame containing a pair of
 13 stereoscopic digital images (L,R) according to a predetermined frame packing format.
 14 For example, in the Accused Instrumentalities, the coded bitstream when it contains a
 15 stereoscopic video in one of the frame packing arrangements such as side-by-side or
 16 top-and-bottom or segmented rectangular frame packing format as defined in the
 17 following sections of the ITU-T H.265 Series H: Audiovisual and Multimedia
 18 Systems, “Infrastructure of audiovisual services – Coding of moving video” High
 19 efficiency video coding (“HEVC Spec”): D.2.16 Frame packing arrangement SEI
 20 message syntax, D.3.16 Frame packing arrangement SEI message semantics, D.2.29
 21 Segmented rectangular frame packing arrangement SEI message syntax, D.3.29
 22 Segmented rectangular frame packing arrangement SEI message semantics. Annex C,
 23 Annex D (Supplemental enhancement information), and Annex E (Video usability
 24
 25
 26
 27
 28

1 information) are each “an integral part of the Recommendation,” i.e., the HEVC Spec.
2 HEVC Spec at Annex C, D, E. The Accused Instrumentalities implement SEI, VUI,
3 and other items as specified in the HEVC Spec, and as stated herein.
4

5 100. The Accused Instrumentalities generate an output video stream which
6 can be reproduced on a visualization apparatus. For example, in the Accused
7 Instrumentalities, the output of the decoding process as defined above is a sequence of
8 decoded pictures. *See, e.g.*, HEVC Spec at 3.39 (“3.39 decoded picture: A decoded
9 picture is derived by decoding a coded picture”). Decoded pictures are the input of
10 the display process. *Id.* at 3.47 (“3.47 display process: A process not specified in this
11 Specification having, as its input, the cropped decoded pictures that are the output of
12 the decoding process.”).
13
14

15 101. The Accused Instrumentalities receive metadata which determine an area
16 occupied by one of the two images within said composite frame, said metadata
17 indicating either a geometry of the frame packing format or a frame packing type of
18 said composite frame. For example, in the Accused Instrumentalities, the HEVC spec
19 provides the default display window parameter to support 2D compatible decoding of
20 stereo formats. *See, e.g.*, HEVC Spec (“NOTE 9 – The default display window
21 parameters in the VUI parameters of the SPS can be used by an encoder to indicate to
22 a decoder that does not interpret the frame packing arrangement SEI message that the
23 default display window is an area within only one of the two constituent frames.”).
24
25
26

27 102. The Accused Instrumentalities determine the area in the composite frame
28

(FC) which is occupied by said one image of the stereoscopic pair within the composite frame based on said metadata. For example, in the Accused Instrumentalities, the default display window parameter has been defined to support this application. The parameter syntax is defined in clause E.2.1 VUI parameters syntax, the semantics thereof being described in clause E.3.1 VUI parameters semantics. The usage of the Default Display Window for signaling the 2D single view in a stereoscopic frame packing format is illustrated in Note 9 of clause D.3.16 and Note 3 in Clause D.3.29 cited above.

103. The Accused Instrumentalities decode only that part of the composite frame which contains said one image to be displayed. For example, in the Accused Instrumentalities, tiles are intended to support independent decoding of different picture regions. Clause 7.4.3.2.1 cited above illustrates the process to convert CTB picture scan in CTB tile scan to enable independent decoding of the tile. *See also* HEVC Spec:

row_height_minus1[i] plus 1 specifies the height of the i-th tile row in units of coding tree blocks.

The following variables are derived by invoking the coding tree block raster and tile scanning conversion process as specified in clause 6.5.1:

- The list CtbAddrRsToTs[ctbAddrRs] for ctbAddrRs ranging from 0 to PicSizeInCtbsY – 1, inclusive, specifying the conversion from a CTB address in the CTB raster scan of a picture to a CTB address in the tile scan,
- the list CtbAddrTsToRs[ctbAddrTs] for ctbAddrTs ranging from 0 to PicSizeInCtbsY – 1, inclusive, specifying the conversion from a CTB address in the tile scan to a CTB address in the CTB raster scan of a picture,
- the list TileId[ctbAddrTs] for ctbAddrTs ranging from 0 to PicSizeInCtbsY – 1, inclusive, specifying the conversion from a CTB address in tile scan to a tile ID,
- the list ColumnWidthInLumaSamples[i] for i ranging from 0 to num_tile_columns_minus1, inclusive, specifying the width of the i-th tile column in units of luma samples,
- the list RowHeightInLumaSamples[j] for j ranging from 0 to num_tile_rows_minus1, inclusive, specifying the height of the j-th tile row in units of luma samples.

The values of ColumnWidthInLumaSamples[i] for i ranging from 0 to num_tile_columns_minus1, inclusive, and RowHeightInLumaSamples[j] for j ranging from 0 to num_tile_rows_minus1, inclusive, shall all be greater than 0.

The array MinTbAddrZs with elements MinTbAddrZs[x][y] for x ranging from 0 to (PicWidthInCtbsY << (CtbLog2SizeY – MinTbLog2SizeY)) – 1, inclusive, and y ranging from 0 to (PicHeightInCtbsY << (CtbLog2SizeY – MinTbLog2SizeY)) – 1, inclusive, specifying the conversion from a location (x, y) in units of minimum transform blocks to a transform block address in z-scan order, is derived by invoking the z-scan order array initialization process as specified in clause 6.5.2.

1 104. The Accused Instrumentalities generate an output frame containing said
2 extracted image. For example, in the Accused Instrumentalities, there is an output of
3 the tile decoding process. *See, e.g.*, HEVC Spec at 8.1.1 (“8.1.1 General...Input to
4 this process is a bitstream. Output of this process is a list of decoded pictures.”).

5
6 105. On information and belief, Defendants also directly infringes and
7 continues to infringe other claims of the ‘298 patent.

8
9 106. On information and belief, all of the Accused Instrumentalities perform
10 the claimed methods in substantially the same way, e.g., in the manner specified in the
11 HEVC standard.

12
13 107. On information and belief, use of the Accused Instrumentalities in their
14 ordinary and customary fashion results in infringement of the methods claimed by the
15 ‘298 patent.

16
17 108. On information and belief, Defendants have had knowledge of the ‘298
18 patent since at least the filing of this Complaint or shortly thereafter, and on
19 information and belief, Defendants knew of the ‘298 patent and knew of its
20 infringement, including by way of this lawsuit. By the time of trial, Defendants will
21 have known and intended (since receiving such notice) that its continued actions
22 would actively induce and contribute to the infringement of the claims of the ‘298
23 patent.
24

25
26 109. Upon information and belief, Defendants’ affirmative acts of making,
27 using, and selling the Accused Instrumentalities, and providing implementation
28

1 services and technical support to users of the Accused Instrumentalities, including,
2 e.g., through training, demonstrations, brochures, installation and user guides, have
3 induced and continue to induce users of the Accused Instrumentalities to use them in
4 their normal and customary way to infringe the '298. For example, Defendants
5 adopted HEVC as its video codec in its products and services. For similar reasons,
6 Defendants also induces its customers to use the Accused Instrumentalities to infringe
7 other claims of the '298 patent. Defendants specifically intended and was aware that
8 these normal and customary activities would infringe the '298 patent. Defendants
9 performed the acts that constitute induced infringement, and would induce actual
10 infringement, with the knowledge of the '298 patent and with the knowledge, or
11 willful blindness to the probability, that the induced acts would constitute
12 infringement. For example, since filing of this action, Defendants know that the
13 ordinary way of using HEVC in the Accused Instrumentalities infringes the patent but
14 nevertheless continues to promote HEVC to customers. The only reasonable
15 inference is that Defendants specifically intend the users to infringe the patent. On
16 information and belief, Defendants engaged in such inducement to promote the sales
17 of the Accused Instrumentalities. Accordingly, Defendants have induced and continue
18 to induce users of the Accused Instrumentalities to use the Accused Instrumentalities
19 in their ordinary and customary way to infringe the '298 patent, knowing that such use
20 constitutes infringement of the '298 patent. Accordingly, Defendants have been (as of
21 filing of the original complaint), and currently is, inducing infringement of the '298
22
23
24
25
26
27
28

1 patent, in violation of 35 U.S.C. § 271(b).

2 110. Defendants have also infringed, and continue to infringe, claims of the
3 ‘298 patent by offering to commercially distribute, commercially distributing, making,
4 and/or importing the Accused Instrumentalities, which are used in practicing the
5 process, or using the systems, of the ‘298 patent, and constitute a material part of the
6 invention. Defendants know the components in the Accused Instrumentalities to be
7 especially made or especially adapted for use in infringement of the ‘298 patent, not a
8 staple article, and not a commodity of commerce suitable for substantial noninfringing
9 use. For example, the ordinary way of using HEVC infringes the patent, and as such,
10 is especially adapted for use in infringement with no substantial noninfringing use.
11 Accordingly, Defendants have been (as of filing of the original complaint), and
12 currently are, contributorily infringing the ‘298 patent, in violation of 35 U.S.C. §
13 271(c).

14 111. By making, using, offering for sale, selling and/or importing into the
15 United States the Accused Instrumentalities, and touting the benefits of using the
16 Accused Instrumentalities’ compression features, Defendants have injured Realtime
17 and is liable to Realtime for infringement of the ‘298 patent pursuant to 35 U.S.C. §
18 271.

19 112. As a result of Defendants’ infringement of the ‘298 patent, Plaintiff
20 Realtime is entitled to monetary damages in an amount adequate to compensate for
21 Defendants’ infringement, but in no event less than a reasonable royalty for the use
22
23
24
25
26
27
28

1 made of the invention by Defendants, together with interest and costs as fixed by the
2 Court.

3
4 **PRAYER FOR RELIEF**

5 WHEREFORE, Plaintiff Realtime respectfully requests that this Court enter:

- 6 a. A judgment in favor of Plaintiff that Defendants have infringed, literally
7 and/or under the doctrine of equivalents the '046, '535, '477, '462, and
8 '298 patents (the "asserted patents");
9
10 b. A judgment and order requiring Defendants to pay Plaintiff its damages,
11 costs, expenses, and prejudgment and post-judgment interest for its
12 infringement of the asserted patents, as provided under 35 U.S.C. § 284;
13
14 c. A judgment and order requiring Defendants to provide an accounting and
15 to pay supplemental damages to Realtime, including without limitation,
16 prejudgment and post-judgment interest;
17
18 d. A permanent injunction prohibiting Defendants from further acts of
19 infringement of the asserted patents;
20
21 e. A judgment and order finding that this is an exceptional case within the
22 meaning of 35 U.S.C. § 285 and awarding to Plaintiff its reasonable
23 attorneys' fees against YouTube; and
24
25 f. Any and all other relief as the Court may deem appropriate and just under
26 the circumstances.
27
28

DEMAND FOR JURY TRIAL

Plaintiff, under Rule 38 of the Federal Rules of Civil Procedure, requests a trial by jury of any issues so triable by right.

Respectfully Submitted,

Dated: April 30, 2018

/s/ Marc A. Fenster

RUSS AUGUST & KABAT
Marc A. Fenster, SBN 181067
Email: mfenster@raklaw.com
Reza Mirzaie (CA SBN 246953)
Email: rmirzaie@raklaw.com
Brian D. Ledahl (CA SBN 186579)
Email: bledahl@raklaw.com
Paul Kroeger (CA SBN 229074)
Email: pkroeger@raklaw.com
C. Jay Chung (CA SBN 252794)
Email: jchung@raklaw.com
Philip X. Wang (CA SBN 262239)
Email: pwang@raklaw.com

Attorneys for Plaintiff

REALTIME ADAPTIVE STREAMING LLC